EAST

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
L1	2	("5734843").PN.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/12 14:47
L2	3	"9913407"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/12 14:50
L3	3	"9315464"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/12 14:50
S1	9137	bus same (divid$3 split$4) same (device$1 component$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/06 11:04
S2	1382	S1 same (capacity width lane$1 channel$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 07:47
S3	259	S2 same (dedicat$3 assign$3 associat$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 07:48
S4	4938	sakashita.in.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 07:32
S5	3	sakashita-yohei.in.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 07:33
S6	2	"11259417"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 07:42
S7	1	1999-595574.NRAN.	DERWENT	OR	OFF	2005/09/15 07:33

S8	1027	virtual adj3 bus	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 07:42
S9	216	virtual adj bus	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 07:42
S10	2921	S1 same (line$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 07:47
S11	482	S10 same (dedicat$3 assign$3 associat$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 08:55
S12	1	2002-640791.NRAN.	DERWENT	OR	OFF	2005/09/15 08:24
S13	91	(sub adj bus$3) with (split$4 divid$3)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 09:30
S14	105	(bus) with (split$4 divid$3) with (portion$1) with (component$1 device$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 09:38
S15	10	("20050014397"|"5862359"|"6172906"|"6182178"|"6502167").PN.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 09:38
S16	4	("5448496" | "5612891" | "6343313" | "6421809").PN.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/09/15 10:11
S17	7	("4357657" | "4589063" | "5089953" | "5319783" | "5781745" | "5784636" | "RE36394").PN.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/09/15 10:13
S18	109	divide adj bus	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 14:43

EAST

S19	1241	bus near5 partition$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 14:54
S20	2	("5,550,990").PN.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/09/15 14:54
S21	49	("5548730" "5649224" "5701515" "5761458" "5768605" "5898894" "5926032" "6035407" "6234829" "6388533" "6389379" "5519715" "5768567" "5771370" "5848264" "6076161" "6212489" "6212489" "4438490" "4458309" "4459656" "4521849" "4525830" "4606003" "4860291" "4954968" "4967326" "5023483" "5243698" "5265101" "5321828" "5353243" "5406607" "5428623" "5502604" "5574894" "5577234" "5587957" "5590354" "5625580" "5687329" "5703622" "5713036" "5764932" "5774793" "5781746" "5826093" "5828674" "5838897" "5838948").pn.	US-PGPUB; USPAT; EPO; JPO; IBM_TDB	OR	OFF	2005/10/03 08:12
S25	2	"11065991"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 08:17
S26	1	1999-237459.NRAN.	DERWENT	OR	OFF	2005/10/03 08:15
S27	8	"1132819"	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 08:40
S28	2002	(711/170,212).CCLS.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 08:52
S29	215	S28 and (bus same (divid$3 subdivid$3 scal$4 split$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 08:52

S30	3099	(710/104,305,307,316).CCLS.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 08:52
S31	801	S30 and (bus same (divid$3 subdivid$3 scal$4 split$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 08:53
S32	533	S30 and (bus with (divid$3 subdivid$3 scal$4 split$4))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 08:54
S33	219	S30 and (bus with (divid$3 subdivid$3 scal$4 split$4) with (device$1 component$1 memor$3 modul$3))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 08:54
S34	7	partial near3 multiple near3 bus	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 09:59
S35	35	ppmb mpmb	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 10:00
S36	865	bus with divid$3 with group$1	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 10:03
S37	235	bus adj5 divid$3 adj5 group$1	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 10:16
S38	571	divid$3 near3 among near3 (device$3 component$1 module$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 10:17
S39	22	bus near3 divid$3 near3 among near3 (device$3 component$1 module$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 10:17
S40	3	("5862359").URPN.	USPAT	OR	OFF	2005/10/03 10:37

EAST

S41	9	("4821174" | "4922409" | "5862359" | "5896543" | "6081863" | "6182178" | "6195723" | "6237057" | "6338107").PN.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/10/03 11:22
S42	4	(virtual adj bus) and myers-p$.xp.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/10/03 11:22
S43	26	("4270167" | "5001625" | "5301283" | "5327538" | "5345566" | "5377189" | "5379394" | "5386517" | "5471587" | "5481679" | "5509127" | "5551007" | "5553249" | "5636361" | "5644733" | "5708784" | "5734850" | "5740174" | "5761448" | "5774684" | "5787265" | "5796964" | "5809533" | "5935232" | "6052752" | "6088753").PN.	US-PGPUB; USPAT; USOCR	OR	OFF	2005/10/03 11:23
S44	1	("6618777").URPN.	USPAT	OR	OFF	2005/10/03 11:25
S45	351	divid$3 with capacity with among	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 13:10
S46	13	S45 with bus	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 13:27
S47	42	logical$3 near3 divid$3 near3 bus	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/03 13:28
S48	2	("20040230732").PN.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2005/10/06 11:04

```
Set     Items   Description
S1     690985   DIVID? OR DIVY? OR ALLOCAT? OR ALLOT? OR DIVVY? OR RATION?
S2     387933   APPORTION? OR DESIGNAT? OR ASSIGN? OR EARMARK? OR DISPENS?
S3     184286   METE? OR PARCEL?()OUT OR DOLE? OR DELEGAT? OR RELEGAT? OR -
            PRORAT? OR PRO()(RATA? OR RATE?)
S4     891218   DISTRIBUT? OR CORRELAT? OR COLLOCAT? OR CONSIGN? OR ROUTE?
S5    3632466   PARTITION? OR SEGMENT? OR PORTION? OR SECTION? OR SUBBUS? -
            OR SUB()(BUS OR BUSES)
S6     267528   DYNAMIC? OR ON(2W)FLY OR REALTIME? OR REAL()TIME? OR WHEN(-
            )NEEDED OR CHANGABL? OR CHANGEABL?
S7      19305   HOTSWAP? OR HOT()SWAP? OR AS()NEEDED? OR CASE(2W)CASE()BAS-
            IS? OR CUSTOMIZ? OR CUSTOMIS?
S8     788795   (CUSTOM? OR TAILOR?)()(MAKE? OR MAKING? OR MADE?) OR CONTI-
            NUOUS? OR CONSTANTLY?
S9     151724   BUS OR BUSSES
S10     13374   PCI? ? OR ISA? ? OR EISA? ? OR VLBUS? OR AGP? ? OR USB? ? -
            OR MCA? ? OR VESABUS? OR VESALOCAL?
S11   1863540   CONNECTOR? OR PIN? ? OR CONDUCTOR? OR WIRE? OR (ELECTRIC? -
            OR ELECTRONIC? OR CIRCUIT? OR IC OR DATA?)(2N)(PATH? OR VIA? ?
             OR CONNECTION?)
S12    132048   (ELECTRIC? OR ELECTRONIC? OR CIRCUIT? OR IC OR DATA?)(2N)(-
            TRACK? OR LANE? OR CHANNEL? OR PORT? OR INTERCONNECT?)
S13   3010556   ACCORDING? OR DEPENDING? OR PER OR ONE(2W)ONE OR CORRESPON-
            D?
S14   1163070   MATCH? OR CONFORM? OR COINCID? OR CONSISTEN? OR SIMILAR? OR
             SYNCHRON?
S15   1026868   EQUAL? OR PROPORTION? OR CONTINGEN? OR HING? OR PREDICAT? -
            OR RELEGAT?
S16    189350   RELY? OR RELIAN? OR COMMENSURAT? OR EQUIVAL? OR COMPARABL?
S17    372339   COMPATIB? OR HOMOLOG? OR SYMMET? OR SYNONYM? OR HOMOGEN?
S18   3198376   NUMBER? OR VOLUME? OR QUANTIT? OR AMOUNT? OR HOW()MANY
S19   1604485   MAGNITUD? OR QUOTA? OR ASSEMBLAG? OR SIZE? OR TOTAL? OR AC-
            CUMULATION?
S20    453527   ABUNDAN? OR DIMENSION?
S21   5650873   ATTACH? OR CONNECT? OR LINK? OR INSTALL? OR PLUG?()IN OR A-
            PPEND?
S22   1354272   COUPL? OR AFFIX? OR HOOK?()UP OR NETWORKED? OR YOKE? OR IN-
            TERFAC?
S23   6978047   DEVICE? OR HARDWARE? OR APPLIANC? OR APPARAT? OR ATTACHMEN-
            T?
S24   2897557   COMPONENT? OR PERIPHERAL? OR PDA? ? OR PALMPILOT? OR PALM(-
            )PILOT? OR ELECTRONIC()PC
S25     17501   (PORTABL? OR MOBIL? OR HANDHELD? OR HAND()HELD? OR ELECTRO-
            NIC? OR PALMTOP? OR PALM()TOP)()(UNIT? ? OR ORGANIZER?)
S26   1245095   IC=G06F?
S27    937807   MC=T01?
S28     66695   S1:S5 AND S9:S12 AND S13:S17 AND S18:S22 AND S23:S25
S29       398   S28 AND S6:S8(7N)S1:S5 AND S1:S8(7N)S9:S12
S30       131   S29 AND S18:S20 AND S21:S22
S31       231   S29 AND S1:S12(7N)S23:S25
S32       103   S29 AND S26:S27
S33       309   S30:S32
S34    797436   PR=2003:2005
S35       280   S33 NOT S34
S36       105   S35 AND S1:S5(7N)S9:S12(10N)S23:S25
S37       119   S30 NOT S34
S38       184   S36:S37
S39       184   IDPAT (sorted in duplicate/non-duplicate order)
S40        53   S32 NOT (S30 OR S36)
S41        44   S40 NOT S34
S42        44   IDPAT (sorted in duplicate/non-duplicate order)
```

File 347:JAPIO Nov 1976-2005/Apr(Updated 050801)
(c) 2005 JPO & JAPIO
File 350:Derwent WPIX 1963-2005/UD,UM &UP=200563
(c) 2005 Thomson Derwent

39/3,K/69 (Item 69 from file: 350)
DIALOG(R)File 350:Derwent WPIX

011813239 **Image available**
WPI Acc No: 1998-230149/199820
XRPX Acc No: N98-182299
Dynamically allocating **bandwidth among** devices **connected by data** bus **with e.g. sixty-four parallel information channels - allocating parts of data bus to devices in response to storage space availability based on fullness of** device **buffer, transmitting data between** devices **on** bus **parts, and providing availability as feedback from buffer to data** bus **controller**
Patent Assignee: ADVANCED MICRO DEVICES INC (ADMI)
Inventor: BELT S L; DUTTON D J; GEPHARDT D D; STEWART B B; WISOR R M
Number of Countries: 001 Number of Patents: 001
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week	
US 5734843	A	19980331	US 95476872	A	19950607	199820	B

Priority Applications (No Type Date): US 95476872 A 19950607
Patent Details:

Patent No	Kind	Lan	Pg	Main IPC	Filing Notes
US 5734843	A		10	G06F-013/00	

Dynamically allocating **bandwidth among** devices **connected by data** bus **with e.g. sixty-four parallel information channels...**

...**of data bus to devices in response to storage space availability based on fullness of** device **buffer, transmitting data between** devices **on** bus **parts, and providing availability as feedback from buffer to data** bus **controller**

...Abstract (Basic): For **allocating** bandwidth among devices connected through a data bus provides, the data need of one of the **devices** is determined and portions of the **data bus** are **allocated** to the **devices** in response to the data need. Data is transmitted between the **devices** on the **allocated portions** of the **data bus** . The **portions** of the **data bus** can be sub- **busses** , each comprising one bit line...

...The data need can be based on a measure of fullness of a buffer **corresponding** to the **device** . The data need can be provided as feedback from the buffer to a data **bus** controller which **allocates** the **portions** of the **data bus** . The method uses rules for **assigning** the **sub - buses** which are stored in a memory. A processor can change the rules to accommodate changing conditions in the data **bus** . A data communication system comprises a **dynamically** reconfigurable data **bus** , a data **bus** controller **connected** to the **dynamically** reconfigurable data **bus** for configuring sub- **busses** of the data **bus** , receiving **devices connected** to the data **bus** , and a feedback **connection** from one of the receiving **devices** to the data **bus** controller. The data **bus** controller configures the sub- **busses** in accordance with feedback received over the feedback **connection** . A memory is **connected** to the data **bus** controller for storing rules for use by the data **bus** controller in configuring the sub- **busses** . A processor changes the rules to accommodate changing conditions in the data **bus** .

...

...ADVANTAGE - **Dynamic allocation** of **bus** access
...Title Terms: **ALLOCATE** ;
International Patent Class (Main): **G06F-013/00**
Manual Codes (EPI/S-X): **T01-H07A2**

US005621901A

United States Patent [19]

Morriss et al.

[11] Patent Number: 5,621,901

[45] Date of Patent: Apr. 15, 1997

[54] **METHOD AND APPARATUS FOR SERIAL BUS ELEMENTS OF AN HIERARCHICAL SERIAL BUS ASSEMBLY TO ELECTRICALLY REPRESENT DATA AND CONTROL STATES TO EACH OTHER**

[75] Inventors: **Jeff C. Morriss**, Boulder Creek; **Andrew M. Volk**, Loomis, both of Calif.

[73] Assignee: **Intel Corporation**, Santa Clara, Calif.

[21] Appl. No.: **332,337**

[22] Filed: **Oct. 31, 1994**

[51] **Int. Cl.**[6] **G06F 13/40**

[52] **U.S. Cl.** **395/306**

[58] **Field of Search** 395/306, 200.13; 341/143

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,245,045	4/1966	Randlev .	
3,916,387	10/1975	Woodrum	340/172.5
4,070,704	1/1978	Calle et al.	395/200.2
4,092,491	5/1978	Frazer	178/67
4,409,656	10/1983	Andersen et al.	395/280
4,606,052	8/1986	Hirzel et al.	375/87
4,660,141	4/1987	Ceccon et al.	395/200.3
4,689,740	8/1987	Moelands et al.	395/200.01
4,748,346	5/1988	Emori	307/270

(List continued on next page.)

OTHER PUBLICATIONS

Philips'I C (Inter–Integrated Circuit) Bus, 5 pages.

Concentration Highway Interface (CHI) AT&T Microelectronics Interface Specification, Nov. 1990 (DS90–124SMOS).

ATA/ANSI 878.1, Version 1.9 (59 Sheets), Copyright 1992 ARCNET Trade Association.

PCMCIA PC Card Standard, Release 2.01, 1.1 –4.8.9, Copyright 1992 PCMCIA.

Access.bus™Specifications –Version 2.2.

High Performance Serial Bus, P1394/Draft 6.2v0, Copyright 1993 IEEE.

Primary Examiner—Ayaz R. Sheikh

Attorney, Agent, or Firm—Blakely, Sokoloff, Taylor & Zafman

[57] **ABSTRACT**

Circuitry and complementary logic are provided to a bus controller, a number of 1:n bus signal distributors, and a number of bus interfaces of an hierarchical bus assembly for electrically representing data and control states each other. The hierarchical serial bus assembly is used to serially interface a number of isochronous and asynchronous peripherals to the system unit of a computer system. The serial bus elements are interconnected to each other using low cost two wire signal cable. Electrical signals are propagated between the serial bus elements in a differential manner. These circuitry and complementary logic of the serial bus elements jointly implement inference of data and control states from the states and/or durations of the propagating electrical signals.

25 Claims, 16 Drawing Sheets

	DATA AND CONTROL STATES	ELECTRICAL REPRESENTATION
706		
708	CONNECT	V+ and V– >Vse0 for 16 bit times
710	DISCONNECT	V+ and V– < Vse0 for 16 bit times
712	START OF PACKET	V+ and V– < Vse0 for 3 bit times
702	END OF PACKET	V+ and V– < Vse0 for 1 bit times
704	LOGICAL "1"	(V+) – (V–) > VDiff
	LOGICAL "0"	(V+) – (V–) < VDiff

METHOD AND APPARATUS FOR SERIAL BUS ELEMENTS OF AN HIERARCHICAL SERIAL BUS ASSEMBLY TO ELECTRICALLY REPRESENT DATA AND CONTROL STATES TO EACH OTHER

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to the field of computer systems. More specifically, the present invention relates to serial buses for connecting peripherals to the system units of computer systems, including the associated controllers and interfaces.

2. Background Information

A number of interrelated considerations is making it desirable to have a single, relatively fast, bi-directional, isochronous, low-cost, and dynamically configurable serial bus for simultaneously connecting isochronous as well as asynchronous peripherals to the system unit of a desktop computer system. Isochronous peripherals are peripherals that generate real time natural data such as voice, motion video, and the like. These interrelated considerations include:

Connection of the Telephone to the Desktop Computer

It is expected that the merging of computing and communication will be the basis of the next generation of productivity applications on desktop computers. The movement of machine oriented and human oriented data types from one location or environment to another depends on ubiquitous and cheap connectivity. Unfortunately, the computing and communication industries have evolved independently. As a result, a wide range of desktop computer and telephone interconnects have to be supported.

Ease of Use

The lack of flexibility in reconfiguring desktop computers has been acknowledged as its Achilles heel to it's further development. The combination of user friendly graphical interfaces and the hardware and software mechanisms associated with the new generation of system bus architectures have made desktop computers less confrontational and easier to reconfigure. However, from the enduser point of view, the desktop computer's I/O interfaces such as serial/parallel ports, keyboard/mouse/joystick interfaces, still lack the attributes of plug and play or too limiting in terms of the type of I/O devices that can be live attached/detached.

Port Expansion

The addition of external peripherals to desktop computers continues to be constrained by port availability. The lack of a bi-directional, low-cost, low to mid speed peripheral bus has held back the proliferation of peripherals like telephone/fax/modem adapters, answering machines, scanners, personal digital assistants (PDA), keyboards, mouses, etc. Existing interconnects are optimized for one or two point products. As each new function or capability is added to the desktop computer, typically a new interface has been defined to address this need.

In other words, this desired serial bus is expected to provide low cost simultaneous connectivity for the relatively low speed 10–100 kbps interactive devices such as keyboard, mouse, stylus, game peripherals, virtual reality peripherals, and monitors, as well as the moderate speed 500–5000 kbps isochronous devices such as ISDN, PBX, POTS, and other audio devices. A multiplicity of both types of devices are expected to be connected and active at the same time, and yet the latter type of devices are provided with guaranteed latencies and bandwidths. Furthermore, the devices are expected to be hot attached and detached, with the serial interface being able to dynamically reconfigure itself without interrupting operation of the desktop computer system.

There are several technologies that are commonly considered to be serial buses for connecting peripherals to system units of computer systems. Each of these buses is designed to handle a specific range of communications between system units and peripherals. Particular examples of these buses include:

Apple® Desktop Bus (ADB): ADB is a proprietary bus of Apple Computer Inc. It is a minimalist serial bus that provides a simple read/write protocol to up to 16 devices. Only basic functions are required of the controller and interface hardware. Thus, the implementation cost is expected to be low. However, ADB supports data rates only up to 90 kbps, just enough to communicate with asynchronous desktop devices such as keyboards and mouses. It is not capable of simultaneously supporting the moderate speed isochronous devices discussed earlier.

Access.bus (A.b): A.b is developed by the Access.bus Industry Group. It is based on the I^2C technology of Philips Corporation and a software model of Digital Equipment Corporation (DEC). A.b is also designed primarily for asynchronous devices such as keyboards and mouses. However A.b is generally considered to be more versatile than ADB. A.b's protocol has well defined specifications for dynamic attach, arbitration, data packets, configuration and software interface. Moderate amount of functions are required of the controller and interface hardware. Thus, the implementation cost is only marginally competitive for the desired desktop application. While addressing is provided for up to 127 devices, the practical loading is limited by cable lengths and power distribution considerations. Revision 2.2 specifies the bus for 100 kbps operation, but the technology has headroom to go up to 400 kbps using the same separate clock and data wires. However, at 400 kbps, A.b still falls short in meeting the requirements of the moderate speed isochronous devices.

IEEE's P1394 Serial Bus Specification (aka FireWire): FireWire is a high performance serial bus. It is designed primarily for hard disk and video peripherals, which may require bus bandwidth in excess of 100 Mbps. It's protocol supports both isochronous and asynchronous transfers over the same set of 4 signal wires, broken up as differential pairs of clock and data signals. Thus, it is capable of simultaneously meeting the requirements of low speed interactive as well as moderate speed isochronous devices. However, elaborate functions are required of the controller and interface hardware, rendering FireWire to be non-price competitive for the desired desktop application. Moreover, the first generation of devices, based on FireWire's specification, are only just becoming available in the market.

The Concentration Highway Interface (CHI): CHI is developed by American Telephone & Telegraph Corporation (AT&T) for terminals and digital switches. It is a full duplex time division multiplexed serial interface for digitized voice transfer in a communication system. The protocol consists of a number of fixed time slots that can carry voice data and control information. The current specification supports data

39/3,K/50 (Item 50 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00775595
HIERARCHICAL SERIAL BUS ASSEMBLY
HIERARCHISCHE SERIELLE BUSVORRICHTUNG
ENSEMBLE DE BUS SERIE HIERARCHIQUE
PATENT ASSIGNEE:
INTEL CORPORATION, (322933), 2200 Mission College Boulevard, Santa Clara, CA 95052, (US), (Proprietor designated states: all)
INVENTOR:
MORRISS, Jeff, Charles, 900 Elsie Mae Drive, Boulder Creek, CA 95006, (US)
VOLK, Andrew, McKenzie, 7380 Sierra Ponds Lane, Loomis, CA 95650, (US)
LEGAL REPRESENTATIVE:
Wombwell, Francis et al (46021), Potts, Kerr & Co. 15, Hamilton Square, Birkenhead Merseyside L41 6BR, (GB)
PATENT (CC, No, Kind, Date): EP 789880 A1 970820 (Basic)
EP 789880 A1 980812
EP 789880 B1 020710
WO 9613778 960509
APPLICATION (CC, No, Date): EP 95939040 951031; WO 95US14245 951031
PRIORITY (CC, No, Date): US 332337 941031
DESIGNATED STATES: DE; GB; NL
INTERNATIONAL PATENT CLASS: **G06F-013/40** ; **G06F-013/38**
NOTE:
No A-document published by EPO
LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	200228	1042
CLAIMS B	(German)	200228	946
CLAIMS B	(French)	200228	1147
SPEC B	(English)	200228	5643
Total word count - document A			0
Total word count - document B			8778
Total word count - documents A + B			8778

INTERNATIONAL PATENT CLASS: **G06F-013/40** ...

... **G06F-013/38**

...SPECIFICATION the field of computer systems. More specifically, the present invention relates to serial buses for **connecting peripherals** to the system units of computer systems, including the associated controllers and **interfaces** .

2. Background Information
A **number** of interrelated considerations is making it desirable to have a single, relatively fast, bi-directional, isochronous, low-cost, and **dynamically configurable** serial **bus** for simultaneously **connecting** isochronous as well as asynchronous **peripherals** to the system unit of a desktop computer system. Isochronous **peripherals** are **peripherals** that generate real time natural data such as voice, motion video, and the like. These interrelated considerations include:

Connection of the Telephone to the Desktop Computer
It is expected that the merging of computing...

...oriented data types from one location or environment to another depends on ubiquitous and cheap **connectivity** . Unfortunately, the computing and communication industries have evolved independently. As a result, a wide range...

...as its Achilles heel to it's further development. The combination of user friendly graphical **interfaces** and the **hardware** and software mechanisms associated with the new generation of system bus architectures have made desktop...

...to reconfigure. However, from the enduser point of view, the desktop computer's I/O **interfaces** such as serial/parallel ports, keyboard/mouse/joystick **interfaces** , still lack the attributes of plug and play or too limiting in terms of the type of I/O **devices** that can be live **attached** /detached.

Port Expansion

The addition of external **peripherals** to desktop computers continues to be constrained by port availability. The lack of a bi-directional, low-cost, low to mid speed **peripheral** bus has held back the proliferation of **peripherals** like telephone/fax/modem adapters, answering machines, scanners, personal digital assistants (**PDA**), keyboards, mouses, etc. Existing interconnects are optimized for one or two point products. As each new function or capability is added to the desktop computer, typically a new **interface** has been defined to address this need.

In other words, this desired serial bus is expected to provide low cost simultaneous **connectivity** for the relatively low speed 10-100 kbps interactive **devices** such as keyboard, mouse, stylus, game **peripherals** , virtual reality **peripherals** , and monitors, as well as the moderate speed 500 - 5000 kbps isochronous **devices** such as ISDN, PBX, POTS, and other audio **devices** . A multiplicity of both types of **devices** are expected to be **connected** and active at the same time, and yet the latter type of **devices** are provided with guaranteed latencies and **bandwidths** . Furthermore, the **devices** are expected to be hot **attached** and detached, with the serial **interface** being able to **dynamically** reconfigure itself without interrupting operation of the desktop computer system.

There are several technologies that are commonly considered to be serial buses for **connecting peripherals** to system units of computer systems. Each of these buses is designed to handle a specific range of communications between system units and **peripherals** . Particular examples of these buses include:

Apple(R) Desktop Bus (ADB): ADB is a proprietary...

...a minimalist serial bus that provides a simple read/write protocol to up to 16 **devices** . Only basic functions are required of the controller and **interface hardware** . Thus, the implementation cost is expected to be low. However, ADB supports data rates only up to 90 kbps, just enough to communicate with asynchronous desktop **devices** such as keyboards and mouses. It is not capable of simultaneously supporting the moderate speed isochronous **devices** discussed earlier.

Access.bus (A.b): A.b is developed by the Access.bus Industry...

...software model of Digital Equipment Corporation (DEC). A.b is also designed primarily for asynchronous **devices** such as keyboards and mouses. However A.b is generally considered to be more versatile than ADB. A.b's protocol has well defined specifications for **dynamic attach** , arbitration, data packets, **configuration** and software **interface** . Moderate **amount** of functions are required of the controller and **interface hardware** . Thus, the implementation cost is only marginally

competitive for the desired desktop application. While addressing is provided for up to 127 **devices** , the practical loading is limited by cable lengths and power **distribution** considerations. Revision 2.2 specifies the **bus** for 100 kbps operation, but the technology has headroom to go up to 400 kbps...

...kbps, A.b still falls short in meeting the requirements of the moderate speed isochronous **devices** .

IEEE's P1394 Serial Bus Specification (aka FireWire):
FireWire is a high performance serial bus. It is designed primarily for hard disk and video **peripherals** , which may require bus **bandwidth** in excess of 100 Mbps. Its protocol supports both isochronous and asynchronous transfers over the...
...of simultaneously meeting the requirements of low speed interactive as well as moderate speed isochronous **devices** . However, elaborate functions are required of the controller and **interface hardware** , rendering FireWire to be non-price competitive for the desired desktop application. Moreover, the first generation of **devices** , based on FireWire's specification, are only just becoming available in the market.

The Concentration Highway **Interface** (CHI): CHI is developed by American Telephone & Telegraph Corporation (AT&T) for terminals and digital switches. It is a full duplex time division **multiplexed** serial **interface** for digitized voice transfer in a communication system. The protocol consists of a **number** of fixed time slots that can carry voice data and control information. The current specification...

...simultaneously meeting the requirements of low speed interactive as well as the moderate speed isochronous **devices** . Similar to FireWire, elaborate functions are also required of the controller and **interface hardware** . As a result, CHI is also non-price competitive for the desired desktop application.
Another...

...below, the present invention provides the desired serial bus assembly, including its associated controller, bridging **connectors** and **interfaces** , that advantageously overcomes the limitations of the prior art serial buses in a novel manner.
According to the present invention there is provided an **apparatus** for incorporation into a first and a second serial bus element of a serial bus assembly having a **plurality** of serial bus elements, to allow the two serial bus elements to electrically represent data and control states to each other, said **apparatus** comprising:
first driver means for incorporation into the first serial bus element for driving a...

...second serial bus elements;
encoding means for incorporation into the first serial bus element and **coupled** to the first driver means for controlling the first driver means thereby encoding data and control states in the first pair of electrical signals, the encoding means encodes a **connected** state by causing said first driver means to drive the first pair of electrical signals...

...outputting difference signals; and
decoding means for incorporation into the second serial bus elements and **coupled** to the second driver means for receiving and decoding the difference signals.
Also according to...

...invention there is provided in a computer system comprising a serial bus assembly having a **plurality** of serial bus elements, a method for the serial bus elements to electrically represent data...

...voltage for a first predetermined duration by the first serial bus element to represent a **connected** state;

b) controlling the electrical signal driving performed in step a) to encode data and...

...illustrates a master/slave model of flow control employed by the present invention for serially **interfacing** the interconnected **peripherals** to the system unit and controlling transaction flows;

Figure 4 illustrates a frame based polling schedule of the present invention implemented by some embodiments for polling the slave " **devices** ";

Figure 5 illustrates geographical and logical addressing of the present invention implemented by some embodiments...

...master/slave model of flow control;

Figure 7 illustrates one embodiment of the cables physically **connecting** the serial bus elements under the present invention;

Figures 8-9 illustrate one embodiment of...

...invention including its associated software;

Figures 10-11 illustrate one embodiment of the 1:n **bus** signal **distributor** of the present invention including its **port circuitry** ;

Figures 12-13 illustrate one embodiment of the bus **interface** of the present invention including its **connector** circuitry;

Figure 14 illustrates one embodiment of the data and control states of the present invention inferable from the electrical signals;

Figure 15 illustrates electrical representations of the " **connected** " and the "disconnected" states under the present invention;

Figure 16 illustrates electrical representations of the...

...end of packet" demarcations under the present invention;

Figure 17 illustrates one embodiment of the **connecting** circuitry provided to the bus controller for implementing electrical representations of the present invention;

Figure 18 illustrates one embodiment of the **port circuitry** provided to the **bus** signal **distributor** for implementing electrical representations of the present invention;

Figure 19 illustrates one embodiment of the **connecting** circuitry provided to the bus **interface** for implementing electrical representations of the present invention.

DETAILED DESCRIPTION

In the following description for purposes of explanation, specific **numbers** , materials and **configurations** are set forth in order to provide a thorough understanding of the present invention. However...form in order not to obscure the present invention.

Hierarchical Serial Bus Assembly for Serially **Interfacing** Isochronous and Asynchronous **Peripherals** to a System Unit of a Computer System

Referring now to Figure 1, a block...

...the present invention is shown. Exemplary computer system 10 comprises system unit 12 having serial **bus** controller 14 of the present invention, 1:n **bus** signal **distributors** 18 of the present invention, each having n+1 ports 24, and **peripherals** 16 having bus **interfaces** 22

of the present invention. **Peripherals** 16 are **coupled** to bus controller 14 of system unit 12 through 1:n **bus** signal **distributors** 18 and cables 20. Collectively, **bus** controller 14, **bus** signal **distributors** 18, **bus interfaces** 22, and cables 20 form a serial **bus** assembly 26 interconnecting bus agents, i.e. system unit 12 and **peripherals** 16 to each other.

Cables 20 are preferably low cost two signal wires cables 48...

...signals are preferably propagated over the two signal wires 48 and 50 between the interconnected **devices** 14, 18 and 22 in a differential manner. For examples, a negative voltage differential represents...

...voltage differential represents a 0-bit. Electrical signals are preferably also used to represent a **number** of control states. A particular implementation of electrically representing data and control states will be...

...functions are well known, and will not be otherwise further described. Similarly, except for bus **interfaces** 22, **peripherals** 16 are intended to represent a broad category of desktop **peripherals** , such as keyboards, mouses, monitors, speakers, microphone, telephones, whose constitutions and functions are also well known, and will not be otherwise further described either. **Bus** controller 14, **bus** signal **distributors** 18 and **bus interfaces** 22 will be described in more detail below with additional references to the remaining figures...

...of Figure 1 in further detail. For this embodiment, serial bus assembly 26' includes serial **bus** controller 14, standalone 1:n **bus** signal **distributor** 18a, integrated 1:n **bus** signal **distributor** 18b, and **bus interfaces** 22a - 22f. The serial **bus** assembly 26' interconnects bus agents telephone 16a, compound keyboard 16b including keyboard, pen and mouse...

...unit 12, the serial bus elements 14, 18a-18b and 22a-22f. and the interconnected **peripherals** 16a-16f form an hierarchy of interconnected **devices** .

Under the present invention, a bus **interface** 22a - 22f is always a termination point. Only a **bus** signal **distributor** , e.g. 18a, may have one or more **bus** signal **distributors** , e.g. 18b, and/or one or more **bus interfaces** , e.g. 16a. **coupled** upstream to it. For the purpose of this disclosure, upstream means "towards the bus controller". Thus, except for the degenerate case where the serial bus assembly 26 has only one **connecting peripheral** 16, typically it is a **bus** signal **distributor** , such as 18a. that is **connected** upstream to the **bus** controller 14.

Furthermore, under the present invention, a **connecting peripheral** may be an isochronous **peripheral** , such as telephone 16a, speakers 16d-16e, and microphone 16f, or asynchronous **peripherals** , such as compound keyboard 16b and monitor 16c. The isochronous **peripherals** may operate with a data transfer rate as high as 5 Mbps, while the asynchronous peripherals may operate with a data transfer rate as high as 100 kbps. Furthermore, a **connecting peripheral** 16a - 16f may be a **multiple** function **peripheral** , i.e. **multiple** functions mapping to a single bus **connection** point serviced by a bus **interface** , e.g. 22b. **Similarly** , although not shown, the system unit 12 may support **multiple** clients.

Figure 3 illustrates a master/slave model of flow control employed by the present invention for serially **interfacing** the interconnected **peripherals** to the system unit and controlling transaction flow. As illustrated, the **bus** controller 14, the signal **bus distributors** 18,

and the **bus interfaces** 22 cooperate to implement the master/slave model of flow control. The **bus** controller 14 serves as the master, and the signal **bus distributors** 18 as well as the **bus interfaces** 22 behave as slave **devices** to the bus controller 14.

Under the master/slave model, the bus controller 14 provides...

...all data communication transactions between the bus agents at their respective operating speeds. The bus **interfaces** 22 engage in data communication transactions on behalf of the functions of the **peripherals** 16. However, the bus **interfaces** 22 accept or transmit data only if they have been authorized or instructed (aka "polled") to do so by the **bus** controller 14. The **bus** signal **distributors** 18 serve strictly as signal **distributors** . They are merely transparent conduits when data communication transactions are conducted by the bus controller 14 and the bus **interfaces** 22 on behalf of the **bus** agents. Thus, the **bus** signal **distributors** 18 never actively participate in data communication transactions, accept data or respond with data.

The **bus** controller 14 systematically polls the functions of the interconnected **peripherals** 16 through their **bus interfaces** 22 for data communication transactions in accordance to a polling schedule which guarantees latencies and **bandwidths** to the isochronous functions of the interconnected **peripherals** 16. Polling of the isochronous functions of the interconnected **peripherals** 16 for data communication transactions are prioritized over for all other polling and as frequent the asynchronous functions of the interconnected **peripherals** 16 through their bus **interfaces** 22 for data communication transactions are scheduled around the polling of the isochronous functions of the interconnected **peripherals** 16 for data communication transactions. Preferably, the polling schedule is dynamically adapted to the interconnected...

...In some embodiments, the bus controller 14, the bus signal distributors 18 and the bus **interfaces** 22 further engage in **connection** management transactions employing the same master/slave model for flow control. **Similarly** , the bus controller 14 provides flow control for conducting the **connection** management transactions at the serial **bus** elements' respective operating speeds. The **bus** signal distributors 18 and the **bus interfaces** 22 respond to the connection management transactions, replying with control/status information as appropriate. The **bus** controller 14, during operation, polls the **bus** signal **distributors** 18 and the **bus** interfaces 22 for such transactions. Polling of the **bus** signal **distributors** 18 and the **bus interfaces** 22 for **connection** management transactions are also scheduled around polling of the isochronous functions of the interconnected **peripherals** 16 for data communication transactions. Preferably, the expanded polling schedule is also **dynamically** adapted to the serial **bus** elements actually present.

Figure 4 illustrate a frame based polling scheduling of the present invention...

...As illustrated, the polling schedule 30, also referred to a super frame, comprises of a **number** of sub-schedules 32, also referred to as soft frames. An isochronous function 34a or 34b of an interconnected **peripheral** 16 is polled as frequent as it is necessary in one or more soft frames 32 of the super frame 30 to guarantee its latency and **bandwidth** . However, an asynchronous function 36a or 36b is polled only once in one soft frame 32 of the super frame 30 for data communication transaction. Similarly, an interconnected **device** 38a or 38b is also polled only once in one soft frame 32 of the super frame 30 for

connection management transaction.

Preferably, all isochronous functions 34a - 34b are polled within a first percentage portion (P1) of a soft frame 32 to ensure the latency and **bandwidth** guaranties are met. Isochronous functions that cannot be accommodated within P1 are preferably rejected for...

...a second percentage portion (P2) of a soft frame 32 to ensure reliability of operation. **Multiple** soft frames 32 are employed if necessary to accommodate all asynchronous function and serial bus...

...polling schedule may be dynamically generated and updated, are described in the copending application, serial **number** 08/331,727, entitled Method And **Apparatus** For Dynamically Generating And Maintaining Frame Based Polling Schedules That Guaranty Latencies And **Bandwidths** To Isochronous Functions.

Figure 5 illustrates geographical and logical addressing of the present invention implemented...

...However, the bus controller 14 is labeled as Host, also referred to as Hub0. The **bus** signal **distributors** 18a-18b are labeled as Hub1 and Hub2. The **peripherals** 16a - 16f including their **corresponding** bus **interfaces** 22a - 22f are jointly labeled as Node0 through Node6. The functions of **peripherals** 16a - 16f are labeled as FN0, FN1, etc.

As illustrated, the serial **bus** elements and functions of the **bus** agents are **assigned** geographical as well as logical addresses (GEO ADDR & LOG ADDR) of a geographical and a...

...be inferred from the GEO ADDR of a Hub 14, and 18a - 18b, and the **connecting** Hub as well as the **connecting** port of the **connecting** Hub may be inferred from the GEO ADDR of a Node 22a - 22f. In one...

...22e are assigned the GEO ADDRes of "Hub1:Port2" and "Hub2:Port3" respectively, identifying the **connecting** Hubs 18a and 18b as "Hub1" and "Hub2" respectively, and the **connecting** ports of **connecting** Hubs 18a and 18b as "Port2" and "Port3" respectively. The functions of Node1 16b are...

...assigned the LOG ADDR of "LA6".

Preferably, the GEO ADDRes and the LOG ADDRes are **dynamically assigned** at power on or reset, and updated in response to live detachment of interconnected **devices** or attachment of additional devices, by the **bus** controller 14 in cooperation with the **bus** signal **distributors** 18 and the **bus** **interfaces** 22. A particular implementation of such **dynamic** **connection** management is described in the copending application, serial **number** 08/332,375, entitled Method And **Apparatus** For **Dynamically** Determining And Managing **Connection** Topology Of An Hierarchical Serial **Bus** Assembly, which is hereby fully incorporated by reference.

For these embodiments, the GEO ADDRes are used to conduct **connection** management transactions among the serial bus element, whereas the LOG ADDRes are used to conduct...

...The separation of the two types of transactions into the two separate address spaces facilitate **dynamic** **connection** management of the serial **bus** elements, without having to interrupt services ...employed to differentiate control packets from data packets. Control packets are packets employed by the **bus** controller 14 to authorize or instruct the **bus** signal **distributors** 18 and the **bus** **interfaces** 22 to engage in transactions. Control packets may also include packets employed by the **bus** signal **distributors** 18 and the **bus** **interfaces** 22 to acknowledge authorizations or instructions from the bus controller 14.

Furthermore, addresses 46 are...

...i.e. "HubX:PortY" or "LAz", may be specified to accommodate those embodiments that support **connection** management transactions and implement both types of addresses.

A particular implementation of employing such communication packets to conduct the various transactions is described in the copending application, serial **number** 08/332.573. entitled Method And **Apparatus** For Exchanging Data, Status And Commands Over An Hierarchical Serial Bus Assembly Using Communication Packets...

...examples, the serial bus elements present their interconnection topology, the functions of the various interconnected **peripherals** , the geographical addresses **assigned** to the serial **bus** elements, the logical addresses **assigned** to the functions of the **interconnected peripherals** . The **data** buffers 56 are used to buffer the data of the data communication transactions between the bus agents. The control state machine and circuitry 52 operates the **hardware** , controlling data communication transactions and employing the above described master/slave model of flow control...

...services 58. For some embodiments, the control state machine and circuitry 52 further operates the **hardware** , controlling **connection** management transactions, implementing the master/slave model of flow control with frame based polling schedule...

...the propagation electrical signals. In particular, the control and state machine circuitry 52 causes the **bus** controller 14 to cooperate with the **bus** signal **distributors** 18 and the **bus interfaces** 22, and perform the electrical representation of data and control states of the present invention...

...machine and circuitry 52 responsive to the operating system 60 and other software such as **device** and **configuring** software 62 and 64 of the system unit 12. In particular, the services include **connection** management such as detection of serial bus elements present, detection of their interconnection topology, detection of the functions of the interconnected **peripherals** , and assignment of the geographical and logical addresses. The services further include transaction management such...

...with functions of the bus agents.

For a more detailed descriptions of the bus controller **hardware** and the bus controller software services 58, refer to the copending applications, serial **numbers** 08/332,375, 08/331,727 and 08/332,573. It should be noted that the **allocation** of functions to the **hardware** and software services of the **bus** controller 14 is implementation dependent. The present invention may be practiced with any **number** of allocations, ranging from minimal **hardware** usage to minimal employment of software services.

Figures 10-11 illustrate one embodiment of the **bus** signal **distributor** of the present invention. The illustrated embodiment is a 1:7 **bus** signal **distributor** 18' having control circuitry 66, control registers 68, and 8 ports 24. Port 0 24 is used to **connect** the **bus** signal **distributor** 18' upstream to the **bus** controller 14 or another **bus** signal **distributor** 18. Ports 1 - 7 are used to **connect** up to a **total** of 7 **bus** signal **distributors** 18 and/or **bus interfaces** 22 to itself. The control registers 68 are used to store its own control and status information such as whether a port 24 has a bus **interface** 22 **connected** to it or not, and whether the port 24 is turned ON/OFF. The control circuitry 66 operates the **bus** signal **distributor** 18'

responsive to instructions from the **bus** controller 14. In particular, the control circuitry 66 causes the **bus** signal **distributor** 18 to cooperate with the **bus** controller 14, and the **bus** **interface** 22 to implement inference of data and control states from states and durations of the...

...72 for generating the differential signals. Preferably, each port 24 further having two resistors 74 **coupled** to ground as shown, pulling the signals on the two wires to ground, thereby allowing the absence or presence of a **connected** bus **interface** 22 to be discernible. The appropriate values of resistors 74 may be determined empirically depending...relationship with the bus controller 14, refer to the incorporated by reference copending application, serial **number** 08/332,375.

Figures 12-13 illustrate one embodiment of the bus **interfaces** of the present invention. For this embodiment, the bus **interface** 22 comprises control circuitry 80, control/status registers 82, a **connector** **interface** 84 and two FIFOs 76 - 78, a Receive FIFO 76 and a Send FIFO 78 ...

...own control and status information such as its assigned geographical address, functions of its "host" **peripheral** , and their **assigned** logical addresses. The control circuitry 66 operates the **bus** **interface** 22 on behalf of the "host" **peripheral** and the "host" **peripheral** 's functions, responsive to authorizations and instructions from the bus controller 14. In particular, the control circuitry 80 causes the bus **interface** 22 to cooperate with the bus controller 14, and the bus **interface** 22 to implement inference of data and control states from states of the propagation electrical...

...the serial bus elements, and electrical signals are preferably propagated in a differential manner, the **connector** **interface** 84 comprises two differential amplifiers 86 and 88 for generating the differential signals. Preferably, the **connector** **interface** 84 further includes two resistors 90 **coupled** to Vcc as shown, pulling the signals on the two wires to Vcc complementary to the **port** **circuitry** of a **connecting** **bus** signal **distributor** 18. The appropriate values of resistors 90 may also be determined empirically depending on individual implementations.

For a more detailed description of the bus **interface** 22, refer to the incorporated by reference copending applications, serial **numbers** 08/332,375, 08/331,727 and 08/332,573.

Electrically Representing Data And Control...

...described the hierarchical serial bus assembly 26 and the manner its element cooperate to serially **interface** the isochronous and asynchronous **peripherals** 16 to the system unit 12 of the exemplary computer system 10, the present invention...

...invention inferable from the electrical signals. Figures 15 and 16 illustrate electrical representations of the " **connected** " and the "disconnected" states, and the "start of packet" and "end of packet" demarcations under...

...greater than a predetermined negative voltage threshold. Furthermore, the data and control states include a " **connected** " state and a "disconnected" state 706 and 708. The " **connected** " state 706 is represented by having the voltages of both electrical signals simultaneously greater than...

...illustrated in Figure 16.

Figures 17,18 and 19 illustrate one embodiment each of the **connecting** circuitry provided to the **bus** controller, the **bus** signal **distributor** , and the **bus interface** for implementing electrical representations of the present invention respectively. The **connection** or port circuitry of each serial **bus** element, i.e. the **bus** controller 14, the **bus** signal **distributor** 18, and the **bus interface** 22 is provided with drivers 720 for outputting the electrical signals (V + & V -) at the...

...the drivers 720 and 722, and decoding the differential signals received.

Thus, a method and **apparatus** for serial bus elements to electrically represent data and control states to each other has...

...invention is not limited to the embodiments described. In particular, beside the low cost two **wires** cable embodiments, the present invention may also be **distributively** practiced over cables with two or more pairs of **wires** . The method and **apparatus** of the present invention can be practiced with modification and alteration within the scope of the **appended** claims. The description is thus to be regarded as illustrative instead of limiting on the...

...CLAIMS B1

1. An **apparatus** for incorporation into a first and a second serial bus element of a serial bus assembly (26, 26') having a **plurality** of serial bus elements, to allow the two serial bus elements (14, 18, 22) to electrically represent data and control states to each other, said **apparatus** comprising:

first driver means (720, 722) for incorporation into the first serial bus element for...

...serial bus elements;

encoding means (726) for incorporation into the first serial bus element and **coupled** to the first driver means for controlling the first driver means thereby encoding data and control states in the first pair of electrical signals, the encoding means encodes a **connected** state by causing said first driver means to drive the first pair of electrical signals...

...difference signals; and

decoding means (728) for incorporation into the second serial bus elements and **coupled** to the second driver means for receiving and decoding the difference signals.

2. The **apparatus** as set forth in claim 1, wherein said encoding means (726) encodes a logical one...

...electrical signals with a positive difference greater than a predetermined positive voltage threshold.

3. The **apparatus** as set forth in claim 1, wherein said encoding means (726) encodes a logical zero...

...a negative difference that is more negative than a predetermined negative voltage threshold.

4. The **apparatus** as set forth in claim 1, wherein said encoding means (726) encodes a "disconnected" state...

...simultaneously smaller than a predetermined single ended voltage (VseO) for a predetermined duration.

5. The **apparatus** as set forth in claim 1, wherein said encoding means (726) encodes a start of...

...simultaneously smaller than a predetermined single ended voltage (VseO) for a predetermined duration.

6. The **apparatus** as set forth in claim 1, wherein said encoding means (726) encodes an end of...

...simultaneously smaller than a predetermined single ended voltage (VseO) for a predetermined duration.

7. The **apparatus** as set forth in claim 1, wherein

said encoding means (726) encodes a " **connected** " state (706) by causing said first driver means (720, 722) to drive the first pair...

...smaller than the predetermined single ended voltage (VseO) for the first predetermined duration.

8. The **apparatus** as set forth in claim 8, wherein

said encoding means (726) encodes a start of...

...duration.

9. In a computer system comprising a serial bus assembly (26, 26') having a **plurality** of serial bus elements, a method for the serial bus elements to electrically represent data...

...VseO) for a first predetermined duration by the first serial bus element to represent a **connected** state;

b) controlling the electrical signal driving performed in step a) to encode data and...

US005742847A

United States Patent [19]

Knoll et al.

[11] Patent Number: 5,742,847

[45] Date of Patent: Apr. 21, 1998

[54] **M&A FOR DYNAMICALLY GENERATING AND MAINTAINING FRAME BASED POLLING SCHEDULES FOR POLLING ISOCHRONOUS AND ASYNCHRONOUS FUNCTIONS THAT GUARANTY LATENCIES AND BANDWIDTHS TO THE ISOCHRONOUS FUNCTIONS**

[75] Inventors: **Shaun Knoll**, Portland, Oreg.; **Jeff Charles Morriss**, Boulder Creek, Calif.; **Ajay V. Bhatt**, El Dorado Hills, Calif.; **Puthiya Kottal Nizar**, El Dorado Hills, Calif.; **Richard M. Haslam**, Hillsboro; **Sudarshan Bala Cadambi**, Portland, both of Oreg.

[73] Assignee: **Intel Corporation**, Santa Clara, Calif.

[21] Appl. No.: **331,727**

[22] Filed: **Oct. 31, 1994**

[51] **Int. Cl.**[6] **G06F 13/14**; G06F 13/20; G06F 13/22

[52] **U.S. Cl.** **395/866**; 395/836; 395/861; 370/95.1; 370/95.2

[58] **Field of Search** 395/835, 849, 395/850, 836, 200.17, 860, 861, 864, 865, 866; 370/60, 60.1, 85.8, 94.2, 95.1, 95.2

[56] **References Cited**

U.S. PATENT DOCUMENTS

3,245,045	4/1966	Randlev	395/800
3,916,387	10/1975	Woodrum	395/800
3,932,841	1/1976	Deerfield et al.	395/287
4,070,704	1/1978	Calle et al.	395/700
4,092,491	5/1978	Frazer	375/283
4,409,656	10/1983	Andersen et al.	331/40
4,606,052	8/1986	Hirzel et al.	375/333
4,660,141	4/1987	Ceccon et al.	395/829
4,689,740	8/1987	Moelands et al.	364/200
4,713,834	12/1987	Brahm et al.	379/28
4,748,346	5/1988	Emori	326/90
4,870,704	9/1989	Matelan et al.	395/800

(List continued on next page.)

OTHER PUBLICATIONS

Philips's I C (Inter-Integrated Circuit) Bus, 5 pages.

Concentration Highway Interface (CHI), AT&T Microelectronics Interface Specification, Nov. 1990 (DS90-124SMOS).

ATA/ANSI 878.1, Version 1.9 (59 Sheets), Copyright 1992 ARCNET Trade Association.

PCMCIA PC Card Standard, Release 2.01, 1.1-4.8.9, Copright 1992 PCMCIA.

ACCESS.bus™ Specifications—Version 2.2.

High Performance Serial Bus, P1394/Draft 6.2v0, Copyright 1993 IEEE.

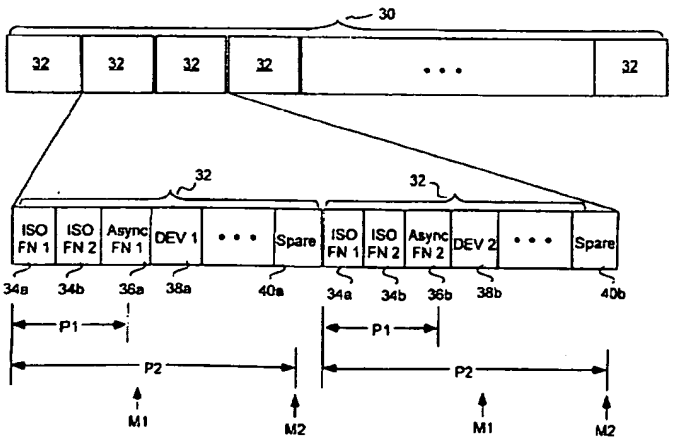
(List continued on next page.)

Primary Examiner—Thomas C. Lee
Assistant Examiner—D. Dinh
Attorney, Agent, or Firm—Blakely, Sokoloff, Taylor & Zafman

[57] **ABSTRACT**

Circuitry and complementary logic are provided to a bus controller, a number of 1:n bus signal distributors, and a number of bus interfaces of an hierarchical serial bus assembly for the bus controller to dynamically generate and maintain a frame based polling schedule for polling the functions of the bus agents connected to the serial bus assembly and the serial bus elements themselves. The hierarchical serial bus assembly is used to serially interface a number of isochronous and asynchronous peripherals to the system unit of a computer system. These circuitry and complementary logic of the serial bus elements support gathering of various critical operating characteristics by the bus controller. The circuitry and logic provided to the bus controller in turn generate the frame based polling schedule in accordance to these gathered critical operating characteristics, guaranteeing latencies and bandwidths to the isochronous functions of the isochronous peripherals. In certain embodiments, the circuitry and logic provided to the bus controller further adapts in real time its frame based polling schedule in like manner, responsive to live attachment/detachment of serial bus elements.

20 Claims, 25 Drawing Sheets



39/3,K/53 (Item 53 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00331266 **Image available**

METHOD AND APPARATUS **FOR DYNAMICALLY GENERATING AND MAINTAINING FRAME BASED POLLING SCHEDULES FOR POLLING ISOCHRONOUS AND ASYNCHRONOUS FUNCTIONS THAT GUARANTEE LATENCIES AND** BANDWIDTHS **TO THE ISOCHRONOUS FUNCTIONS**

PROCEDE ET APPAREIL DE GENERATION ET MAINTENANCE DYNAMIQUES DE CALENDRIERS DE CONSULTATION AU NIVEAU DE LA TRAME EN VUE DE CONSULTER DES FONCTIONS ISOCHRONES ET ASYNCHRONES QUI ASSURENT DES TEMPS D'ATTENTE ET DES LARGEURS DE BANDE AUX FONCTIONS ISOCHRONES

Patent Applicant/Assignee:
INTEL CORPORATION,
Inventor(s):
KNOLL Shaun,
MORRISS Jeff Charles,
BHATT Ajay V,
NIZAR Puthiya Kottal,
HASLAM Richard M,
CADAMBI Sudarshan Bala,

Patent and Priority Information (Country, Number, Date):
Patent: WO 9613777 A1 19960509
Application: WO 95US14209 19951030 (PCT/WO US9514209)
Priority Application: US 94331727 19941031
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
AL AM AT AU BB BG BR BY CA CH CN CZ CZ DE DE DK DK EE EE ES FI FI GB GE HU IS JP KE KG KP KR KZ LK LR LS LT LU LV MD MG MK MN MW MX NO NZ PL PT RO RU SD SE SG SI SK SK TJ TM TT UA UG UZ VN KE LS MW SD SZ UG AT BE CH DE DK ES FR GB GR IE IT LU MC NL PT SE BF BJ CF CG CI CM GA GN ML MR NE SN TD TG
Publication Language: English
Fulltext Word Count: 12599

METHOD AND APPARATUS **FOR DYNAMICALLY GENERATING AND MAINTAINING FRAME BASED POLLING SCHEDULES FOR POLLING ISOCHRONOUS AND ASYNCHRONOUS FUNCTIONS THAT GUARANTEE LATENCIES AND** BANDWIDTHS **TO THE ISOCHRONOUS FUNCTIONS**

Main International Patent Class: **G06F-013/12**
Fulltext Availability:
Detailed Description
Claims

English Abstract
Circuitry and complementary logic are provided to a **bus** controller (14), a **number** of 1:n **bus** signal **distributors** (18a-18b), and a **number** of **bus interfaces** (22a-22f) of an hierarchical serial bus assembly (26') for the **bus** controller (14) to **dynamically** generate and maintain a frame based polling schedule for polling the functions of the bus agents (16a-16f) **connected** to the serial bus assembly (26') and the serial **bus** elements themselves. The hierarchical serial **bus** assembly (26') is used to serially **interface** a **number** of isochronous (16a, 16d-16f) and asynchronous (16b-16c) **peripherals** to the system unit of a computer system. The circuitry and complementary logic of the
...

...circuitry and logic provided in accordance to these gathered critical

operating characteristics guarantee latencies and **bandwidths** to the isochronous functions of the isochronous (16a, 16d-16f) **peripherals** . In certain embodiments, the circuitry and logic provided to the **bus** controller (14) further adapt in **real time** its frame based polling schedule in like manner, responsive to live **attachment** /detachment of serial bus elements.

French Abstract

Un circuit principal et une logique complementaire desservent un controleur de **bus** (14), une **quantite** de 1:n **distributeurs** de signaux de **bus** (18a-18b) et une **quantite** d' **interfaces** de bus (22a-22f) d'un ensemble de bus serie hierarchique (26') de facon a...

...calendrier de consultation en vue de consulter les fonctions des agents du bus (16a-16f) **connectes** a l'ensemble de bus serie (26') et aux elements eux-memes de bus serie. L'ensemble de bus serie hierarchique (26') est utilise pour relier en serie une **quantite** de peripheriques isochrones (16a, 16d-16f) et asynchrones (16b-16c) a l'unite d'un...

Detailed Description

AD

Method 4& **Apparatus** For Dynamically G knd Maintaining Frame Based Polling Schedules For Polling Isochronous And Asynchronous Functions That Guaranti Latencies And **Bandwidths** To The Isochronous Functions

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present...

...the field of computer systems. More specifically, the present invention relates to serial buses for **connecting peripherals** to the system units of computer systems, including the associated controllers and **interfaces** .

2. Background Information

A **number** of interrelated considerations is making it desirable to have a single, relatively fast, bi-directional, isochronous, low-cost, and **dynamically configurable** serial **bus** for simultaneously **connecting** isochronous as well as asynchronous **peripherals** to the system unit of a desktop computer system. Isochronous **peripherals** are **peripherals** that generate real time natural data such as voice, motion video, and the like. These interrelated considerations include.

Connection of the Telephone to the Desktop Computer

It is expected that the merging of computing...

...as its Achilles heel to it's further development. The combination of user friendly graphical **interfaces** and the **hardware** and software mechanisms associated with the new generation of system bus architectures have made desktop...

...to reconfigure. However, from the enduser point of view, the desktop computer's I/O **interfaces** such as serial/parallel ports, keyboard/mouse/joystick **interfaces** , still lack the attributes of plug and play or too limiting in terms of the type of I/O **devices** that can be live **attached** /detached.

Port Expansion

The addition of external **peripherals** to desktop computers continues to be constrained by port availability. The lack of a bi-directional, low-cost, low to mid speed **peripheral** bus has held back the proliferation of **peripherals** like telephone/fax/modem adapters, answering machines, scanners, personal digital assistants (**PDA**), keyboards, mouses, etc. Existing interconnects are optimized for one or two point products. As each new function or capability is added to the desktop computer, typically a new **interface** has been defined to address this need.

In other words, this desired serial bus is...

...to provide low cost simultaneous connectivity for the relatively low speed 10-100 kbps interactive **devices** such as keyboard, mouse, stylus, game **peripherals** , virtual reality **peripherals** , and monitors, as well as the moderate speed 500 - 5000 kbps isochronous **devices** such as ISDN, PBX, POTS, and other audio **devices** . A multiplicity of both types of **devices** are expected to be **connected** and active at the same time, and yet the latter type of **devices** are provided with guaranteed latencies and **bandwidths** . Furthermore, the **devices** are expected to be hot **attached** and detached, with the serial **interface** being able to **dynamically** reconfigure itself without interrupting operation of the desktop computer system.

There are several technologies that are commonly considered to be serial buses for **connecting** **peripherals** to system units of computer systems. Each of these buses is designed to handle a specific range of communications between system units and **peripherals** . Particular examples of these buses include.

A12121e@ Desktop Bus (ADB): ADB is a proprietary bus...

...a minimalist serial bus that provides a simple read/write protocol to up to 16 **devices** . Only basic functions are required of the controller and **interface** **hardware** . Thus, the implementation cost is expected to be low. However, ADB supports data rates only up to 90 kbps, just enough to communicate with asynchronous desktop **devices** such as keyboards and mouses. It is not capable of simultaneously supporting the moderate speed isochronous **devices** discussed earlier.

Access.bus (Ab): A.b is developed by the Access.bus Industry Group...

...software model of Digital Equipment Corporation (DEC). A.b is also designed primarily for asynchronous **devices** such as keyboards and mouses. However A-b is generally considered to be more versatile than ADB. A.b's protocol has well defined specifications for **dynamic** **attach** , arbitration, data packets, **configuration** and software **interface** . Moderate **amount** of functions are required of the controller and **interface** **hardware** . Thus, the implementation cost is only marginally competitive for the desired desktop application. While addressing is provided for up to 127 **devices** , the practical loading is limited by cable lengths and power **distribution** considerations. Revision 2.2 specifies the **bus** for 100 kbps operation, but the technology has headroom to go up to 400 kbps...

...kbps, A.b still
falls short in meeting the requirements of the moderate speed isochronous **devices** .

IEEE's P1394 Serial Bus Specification (aka FireWire).

FireWire is a high performance serial bus. It is designed primarily for hard disk and video **peripherals** , which may require bus **bandwidth** in excess of 100 Mbps. It's protocol supports both isochronous and asynchronous transfers over...

...of
simultaneously meeting the requirements of low speed interactive as well as moderate speed isochronous **devices** . However, elaborate functions are required of the controller and **interface hardware** , rendering FireWire to be non-price competitive for the desired desktop application. Moreover, the first generation of **devices** , based on FireWire's specification, are only just becoming available in the market.

The Concentration Highway **Interface** (CHI): CHI is developed by American Telephone & Telegraph Corporation (AT&T) for terminals and digital switches. It is a full duplex time division multiplexed serial **interface** for digitized voice transfer in a communication system. The protocol consists of a **number** of fixed time slots that can carrv voice data and control information. The current specification...

...simultaneously
meeting the requirements of low speed interactive as well as the moderate speed isochronous **devices** . Similar to FireWire, elaborate functions are also required of the controller and **interface hardware** .

As a result, CHI is also non-price competitive for the desired desktop application.

As...

...present
invention provides the desired serial bus assembly, including its associated controller, bridging connectors and **interfaces** , that advantageously overcomes the limitations of the prior art serial buses in a novel manner.

SUMMARY OF THE INVENTION
The present invention includes circuitry and complementary logic provided to a **bus** controller, a **number** of 1:n bus signal **distributors** , and a **number** of **bus interfaces** of an hierarchical
serial **bus** assembly for the **bus** controller to **dynamically** generate and
maintain a frame based polling schedule for polling the functions of the bus agents **connected** to the serial **bus** assembly and the serial **bus**
elements themselves. The hierarchical serial **bus** assembly is used to serially **interface** a **number** of isochronous and asynchronous **peripherals** to the system unit of a computer system. These circuitry and complementary logic of the...

...frame based polling schedule in accordance to these gathered critical

operating characteristics, guaranteeing latencies and **bandwidths** to the isochronous functions of the isochronous **peripherals** . In certain embodiments, the circuitry and logic provided to the bus controller further adapt in...

...elements.

Typically, the bus controller is disposed in the system unit, and interconnected to a **number** of I:n bus signal distributors and a **number** of bus **interfaces** . The **bus interfaces** are disposed in the **connecting** peripherals, one **bus interface per connecting** peripheral. The peripherals, through their **bus interfaces** , are generally **connected** to the system unit, through the **bus** controller, using one or more of the **bus** signal distributors disposed in the system unit, stand alone bridging connectors and/or the connecting **peripherals** . A **bus interface** is always a termination point. Only a **bus** signal **distributor** may have one or more **bus** signal **distributor** (s) and/or **bus interface** (s) **connected** upstream to it. Together the svstem unit, the serial bus elements, and the **peripherals** form an hierarchy of interconnected **devices** .

A **connecting peripheral** may be an isochronous or an asynchronous **peripheral** . Typically, the isochronous **peripherals** operate with data rates in the range of 500 - 5000 kbps, whereas the asynchronous **peripherals** operate with data rates in the range of 10 100 kbps. Furthermore, a **connecting** peripheral may be a multi function peripheral, i.e. **multiple** functions being mapped to a single **bus connection** point serviced by a single **bus interface** .

The **bus** controller, the **bus** signal **distributors** , and the **bus** interfaces jointly implement a master/slave model of flow control for serially interfacing the interconnected **peripherals** to the system unit to facilitate data communication transactions between the bhs agents, and **connection** management transactions among the serial bus elements at the bus agents and the serial bus...

...elements jointly implement a frame based polling schedule for polling the functions of the slave **peripherals** and the slave serial bus elements that guaranties the latencies and **bandwidths** of the isochronous functions. The frame based schedule, also referred to as a super frame, has a **number** of sub schedules, also referred to as soft frames. An isochronous function is polled as...

...in one or more soft frames of the super frame to guarantee its latency and **bandwidth** . The rest of the pollings are scheduled around the polling of the isochronous functions, typically...

...one soft frame. During operation, the bus controller systematically polls the functions of the interconnected **peripherals** through their bus **interfaces** for data communication transactions, and the serial bus elements for **connection** management transactions in accordance to the polling schedule.

At power on or reset, the circuitry...

...functions, transmit times of their isochronous data, transactional requirements for asynchronous functions as well as **connection** management, and transmit times of these transactions.

The circuitry and logic provided to the bus...

...then determine the lowest latency requirement of the isochronous functions, and set the soft frame **size** to equal the lowest latency requirement of the isochronous functions. These circuitry and logic of...

...polling of the asynchronous functions for data communication transactions, and the serial bus elements for **connection** management transactions, around the polling of the isochronous functions for data communication transactions, subject to frame watermark, using **multiple** soft frames if necessary. In other embodiments, the circuitry and logic provided to the bus...

...and logic provided to the bus controller then schedule polling of the isochronous functions in **multiple** soft frames, for data communication transactions, based on truncated versions of these multipliers, and subject...

...Polling of the asynchronous functions for data communication transactions, and the serial bus elements for **connection** management transactions, are scheduled around the polling of the isochronous functions for data communication transactions as the earlier described approach.

In some embodiments, the **bus** controller, the bus signal **distributors** , and the **bus** interfaces also jointly support having the **bus** controller **dynamically** detects live **attachment** /detachment of serial **bus** elements and **dynamically** manages the **connection** topology. For these embodiments, the circuitry and logic provided to the **bus** controller further adapt in **real time** the frame based polling schedule in like manner.

BRIEF DESCRIPTION OF DRAWINGS

The present invention...

...illustrates a master/slave model of flow control employed by the present invention for serially **interfacing** the interconnected **peripherals** to the system unit and controlling transaction flows;
Figure 4 illustrates a frame based polling schedule of the present invention implemented by some embodiments for polling the slave " **devices** ";
Figure 5 illustrates geographical and logical addressing of the present invention implemented by some embodiments...

...master/slave model of flow control;
Figure 7 illustrates one embodiment of the cables physically **connecting** the serial bus elements under the present

invention;
Figures 8-9 illustrate one embodiment of...

...invention including its associated software;
Figures 10-11 illustrate one embodiment of the 1:n **bus** signal **distributor** of the present invention including its **port circuitry** ;
Figures 12-13 illustrate one embodiment of the bus **interface** of the present invention including its connector circuitry;
Figure 14 illustrates the lowest latency method of the present invention for generating a frame based polling schedule that guanraties latencies and **bandwidths** of isochronous functions;
Figures 15a-15c illustrate the method steps of the present invention for...

...of the present invention for generating a frame based polling schedule that guartraties latencies and **bandwidths** of isochronous functions;
Figures 19a-19c illustrate the method steps of the present invention for...

...common update rate method.

DETAILED DESCRIPTION
In the following description for purposes of explanation, specific **numbers** , materials and **configurations** are set forth in order to
provide a thorough understanding of the present invention.

However...

...form in order not to obscure the present invention.

Hierarchical Serial Bus Assembly for Serially **Interfacing** Isochronous and Asyncrhnous **Peripherals** to a System Unit of a Computer System
Referring now to Figure 1, a block...

...the present invention is shown. Exemplary computer system 10 comprises system unit 12 having serial **bus** controller 14 of the present invention, 1:n **bus** signal **distributors** 18 of the present invention, each
having n+1 ports 24, and **peripherals** 16 having bus **interfaces** 22 of the
present invention. **Peripherals** 16 are **coupled** to bus controller 14 of system unit 12 through 1:n **bus** signal **distributors** 18 and cables 20.

Collectively, **bus** controller 14, **bus** signal **distributors** 18, **bus interfaces**
22, and cables 20 form a serial **bus** assembly 26 interconnecting bus agents, Le. system unit 12 and **peripherals** 16 to each other.

Cables 20 are preferably low cost two signal wires cables 48...

...signals are preferably propagated over the two signal wires 48 and 50 between the interconnected **devices** 14, 18 and 22 in ...voltage states and/or signal durations is described in the
contemporaneously filed copending application, serial **number** 08/332,337 entitled Method And **Apparatus** For Serial Bus Elements Of An Hierarchical Serial Bus To Electrically Represent Data And Control...

...functions are well known, and will not be otherwise further described. Similarly, except for bus **interfaces** 22, **peripherals** 16 are intended to represent a broad category of desktop **peripherals** , such as keyboards, mouses, monitors, speakers, microphone, telephones, whose constitutions and functions are also well known, and will not be otherwise further described either. **Bus** controller 14, **bus** signal **distributors** 18 and **bus** **interfaces** 22 will be described in more detail below with additional references to the remaining figures...

...of Figure 1 in further detail. For this embodiment, serial bus assembly 26' includes serial **bus** controller 14, standalone I:n **bus** signal **distributor** 18a, integrated I:n **bus** signal **distributor** 18b, and **bus** **interfaces** 22a - 22f. The serial **bus** assembly 26' interconnects bus agents telephone 16a, compound keyboard 16b including keyboard, pen and mouse...

...unit 12, the serial bus elements 14, 18a-18b and 22a-22f,, and the interconnected **peripherals** 16a-16f form an hierarchy of interconnected **devices** .

Under the present invention, a bus **interface** 22a - 22f is always a termination point. Only a **bus** signal **distributor** , e.g. 18a, may have one or more **bus** signal **distributors** , e.g. 18b, and/or one or more **bus** **interfaces** , e.g. 16a, **coupled** upstream to it. For the purpose of this disclosure, upstream means "towards the bus controller". Thus, except for the degenerate case where the serial bus assembly 26 has only one **connecting** **peripheral** 16, typically it is a **bus** signal **distributor** , such as 18a, that is **connected** upstream to the **bus** controller 14.

Furthermore, under the present invention, a **connecting** **peripheral** may be an isochronous **peripheral** , such as telephone 16a, speakers 16d-16e, and microphone 16f, or asynchronous **peripherals** , such as compound keyboard 16b and monitor 16c. The isochronous **peripherals** may operate with a data transfer rate as high as 5 Mbps, while the asynchronous peripherals may operate with a data transfer rate as high as 100 kbps. Furthermore, a **connecting** **peripheral** 16a 16f may be a **multiple** function **peripheral** , i.e. **multiple** functions mapping to a single bus **connection** point serviced by a bus **interface** , e.g. 22b. **Similarly** , although not shown, the system unit 12 may support **multiple** clients.

Figure 3 illustrates a master/slave model of flow control employed by the present invention for serially **interfacing** the interconnected **peripherals** to the system unit and controlling transaction flow. As illustrated, the **bus** controller 14, the signal **bus** **distributors** 18, and the **bus** **interfaces** 22 cooperate to implement the master/slave model of flow control. The **bus** controller 14 serves as the master, and the signal **bus** **distributors** 18 as well as the **bus**

interfaces 22 behave as slave **devices** to the bus controller 14.

Under the master/slave model, the bus controller 14 provides...

...all data communication transactions between the bus agents at their respective operating speeds. The bus **interfaces** 22 engage in data communication transactions on behalf of the functions of the **peripherals** 16. However, the bus **interfaces** 22 accept or transmit data only if they have been authorized or instructed (aka "polled") to do so by the **bus** controller 14. The **bus** signal **distributors** 18 serve strictly as signal **distributors** . They are merely transparent conduits when data communication transactions are conducted by the bus controller 14 and the bus **interfaces** 22 on behalf of the **bus** agents. Thus' the **bus** signal **distributors** 18 never actively participate in data communication transactions, accept data or respond with data.

The **bus** controller 14 systematically polls the functions of the interconnected **peripherals** 16 through their **bus** **interfaces** 22 for data communication transactions in accordance to a polling schedule which guarantees latencies and **bandwidths** to the isochronous functions of the interconnected **peripherals** 16. Polling of the isochronous functions of the interconnected **peripherals** 16 for data communication transactions are prioritized over for all other polling and as frequent...

...they are necessary to meet the guaranty. Polling of the asynchronous functions of the interconnected **peripherals** 16 through their bus **interfaces** 22 for data communication transactions are scheduled around the polling of the isochronous functions of the interconnected **peripherals** 16 for data communication transactions.

Preferably, the polling schedule is dynamically adapted to the interconnected...

...In some embodiments, the bus controller 14, the bus signal distributors 18 and the bus **interfaces** 22 further engage in **connection** management transactions employing the same master/slave model for flow control. **Similarly** , the bus controller 14 provides flow control for conducting the **connection** management transactions at the serial **bus** elements' respective operating speeds.

The **bus** signal distributors 18 and the **bus** **interfaces** 22 respond to the connection management transactions, replying with control/status information as appropriate. The **bus** controller 14, during operation, polls the **bus** signal **distributors** 18 and the **bus** interfaces 22 for such transactions. Polling of the **bus** signal **distributors** 18 and the **bus** **interfaces** 22 for **connection** management transactions are also scheduled around polling of the isochronous functions of the interconnected **peripherals** 16 for data communication transactions.

Preferably, the expanded polling schedule is also **dynamically** adapted to the serial **bus** elements actually present.

...As illustrated, the polling schedule 30, also

referred to a super frame, comprises of a **number** of sub-schedules 32, also referred to as soft frames. An isochronous function 34a or 34b of an interconnected **peripheral** 16 is polled as frequent as it is necessary
in one or more soft frames 32 of the super frame 30 to guarantee its latency and **bandwidth** . However, an asynchronous function 36a or 36b is polled only once in one soft frame 32 of the super frame 30 for data communication transaction. Similarly, an interconnected **device** 38a or 38b is also polled only once in one soft frame 32 of the super frame 30 for **connection** management transaction.

Preferably, all isochronous functions 34a - 34b are polled within a first percentage portion (Pl) of a soft frame 32 to ensure the latency and **bandwidth** guaranties are met. Isochronous functions that cannot be accommodated within P1 are preferably rejected for...

...a second percentage portion (P2) of a soft frame 32 to ensure reliability of operation. **Multiple** soft frames 32 are employed if necessary to accommodate all asynchronous function and serial bus...

...However, the bus controller 14 is labeled as Host, also referred to as HubO. The **bus** signal **distributors** 18a-18b are labeled as
Hubl and Hub2. The **peripherals** 16a - 16f including their **corresponding** bus **interfaces** 22a - 22f are jointly labeled as NodeO through Node6. The functions of **peripherals** 16a - 16f are labeled as FNO, FN1, etc.

As illustrated, the serial **bus** elements and functions of the **bus** agents are **assigned** geographical as well as logical addresses (GEO ADDR & LOG ADDR) of a geographical and a...

...be inferred from the GEO ADDR of a Hub 14, and 18a - 18b, and the **connecting** Hub as well as the **connecting** port of the **connecting** Hub may be inferred from the GEO ADDR of a Node 22a - 22f. In one...and 22e are assigned the GEO ADDRes of "Hubl:Port2" and "HuUPortY respectively, identifying the **connecting** Hubs 18a and 18b as "Hubl" and "Hub2" respectively, and the **connecting** ports of **connecting** Hubs 18a and 18b as "Port2" and "PortY respectively. The functions of Nodel 16b are...

...assigned the LOG ADDR of
11LA611
Preferably, the GEO ADDRes and the LOG ADDRes are **dynamically assigned** at power on or reset, and updated in response to live detachment of interconnected **devices** or attachment of additional devices, by the **bus** controller 14 in cooperation with the **bus** signal **distributors** 18 and the **bus interfaces** 22. A particular implementation
of such **dynamic connection** management is described in the copending application, serial **number** 08/332,375, entitled Method And **Apparatus** For **Dynamically** Determining And Managing **Connection** Topology Of An Hierarchical Serial **Bus** Assembly, which is hereby fully incorporated by reference.

For these embodiments, the GEO ADDRes are used to conduct **connection** management transactions among the serial bus element, whereas the LOG ADDRes are used to conduct...

...The separation of the two types of transactions into the two separate address spaces facilitate **dynamic connection** management of the serial **bus** elements, without having to interrupt services to the functions of the bus agents.
Figure 6...

...employed to differentiate
control packets from data packets. Control packets are packets employed by the **bus** controller 14 to authorize or instruct the **bus** signal **distributors** 18 and the **bus interfaces** 22 to engage in transactions. Control packets may also include packets employed by the **bus** signal **distributors** 18 and the **bus interfaces** 22 to acknowledge
authorizations or instructions from the bus controller 14.

Furthermore, addresses 46 are...

...i.e. "HubX:PortY" or "LAz", may be specified to
accommodate those embodiments that support **connection** management transactions and implement both types of addresses.

A particular implementation of employing such
communication packets to conduct the various transactions is described in the contemporaneously filed copending application, serial **number** 08/332,573, entitled Method And **Apparatus** For Exchanging Data, Status And Commands Over An Hierarchical Serial Bus Assembly Using Communication Packets...

...examples, the serial bus elements
present, their interconnection topology, the functions of the various interconnected **peripherals** , the geographical addresses **assigned** to the
serial **bus** elements, the logical addresses **assigned** to the functions of
the **interconnected peripherals** . The **data** buffers 56 are used to buffer
the data of the data communication transactions between the bus agents. The control state machine and circuitry 52 operates the **hardware** , controlling data communication transactions and employing the above described master/slave model of flow control...

...services 58. For
some embodiments, the control state machine and circuitry 52 further operates the **hardware** , controlling **connection** management transactions, implementing the master/slave model of flow control with frame based polling schedule...

...the propagation electrical signals. In particular, the control and state mahchine circutiry 52 causes the **bus** controller 14 to cooperate with the **bus** signal **distributors** 18 and the **bus interfaces** 22, and perform
the frame based polling schedule generation and maintenance steps of the present...

...machine and circuitry 52 responsive to the operating
system 60 and other software such as **device** and **configuring** software 62 and 64 of the system unit 12. In particular, the services include **connection** management such as detection of serial bus elements present, detection of their interconnection topology, detection of the

functions of the interconnected **peripherals** , and assignment of the geographical and logical addresses. The services further include transaction management such...

...with
functions of the bus agents.

For a more detailed description of the bus controller **hardware** and the bus controller software services 58, refer to the incorporated by reference copending applications, serial **numbers** 08/332,375, 08/332,573 and 08/332,337. It should be noted that the **allocation** of functions to the **hardware** and software services of the **bus** controller 14 is implementation dependent. The present invention may be practiced with any **number** of allocations, ranging from minimal **hardware** usage to minimal employment of software services.

Figures 10-11 illustrate one embodiment of the **bus** signal **distributor** of the present invention. The illustrated embodiment is a 1:7 **bus** signal **distributor** 18' having control circuitry 66, control registers 68, and 8 ports 24. Port 0 24 is used to **connect** the **bus** signal **distributor** 18' upstream to the **bus** controller 14 or another **bus** signal **distributor** 18. Ports 1 - 7 are used to **connect** up to a **total** of 7 **bus** signal **distributors** 18 and/or **bus** **interfaces** 22 to itself. The control registers 68 are used to store its own control and status information such as whether a port 24 has a bus **interface** 22 **connected** to it or not, and whether the port 24 is turned ON/OFF. The control circuitry 66 operates the **bus** signal **distributor** 18' responsive to instructions from the **bus** controller 14. In particular, the control circuitry 66 causes the **bus** signal **distributor** 18 to cooperate with the **bus** controller 14, and perform the frame based polling schedule generation and maintenance steps of the...

...72 for
generating the differential signals. Preferably, each port 24 further having two resistors 74 **coupled** to ground as shown, pulling the signals on the two wires to ground, thereby allowing the absence or presence of a **connected** bus **interface** 22 to be discernible. The appropriate values of resistors 74 may be determined empirically depending on individual implementations.

For a more detailed description of the **bus** signal **distributor** 18, refer to the incorporated by reference copending application, serial **number** 08/332,375.

Figures 12-13 illustrate one embodiment of the bus **interfaces** of the present invention. For this embodiment, the bus **interface** 22 comprises control circuitry 80, control/status registers 82, a Connector **interface** 84 and two FIFOs 76 - 78, a Receive FIFO 76 and a Send FIFO 78...

...own control and status
information such as its assigned geographical address, functions of its "host" **peripheral** , and their **assigned** logical addresses. The control circuitry 66 operates the **bus interface** 22 on behalf of the "host" **peripheral** and the "host" **peripheral** 's functions, responsive to authorizations and instructions from the bus controller 14. In particular, control circuitry 66 causes the bus **interface** 22 to cooperate
with the bus controller 14, and perform the frame based polling schedule differential manner,
the Connector **interface** 84 comprises two differential amplifiers 86 and 88 for generating the differential signals. Preferably, the Connector **interface** 84 further includes two resistors 90 **coupled** to Vcc
as shown, pulling the signals on the two wires to Vcc complementary to the **port circuitry** of a **connecting bus** signal **distributor** 18. The
appropriate values of resistors 90 may also be determined empirically depending on individual implementations.

t
For a more detailed description of the bus **interface** 22, refer to the incorporated by reference copending applications, serial **number** 08/332,375, 08/332,573 and 08/332,337.

Frame Based Polling Schedule Generation...

...described the hierarchical serial bus
assembly 26 and the manner its elements cooperate to serially **interface** the isochronous and asynchronous **peripherals** 16 to the system unit 12 of the exemplary computer system 10, the frame based...

...of the
present invention for generating a frame based polling schedule that guaranties latencies and **bandwidths** of isochronous functions. As illustrated, at power on or reset, the bus controller 14, in...

...elements, determines the latency
requirement of each isochronous function, step 302, sets the soft frame **size** to the lowest latency requirement, step 304, computes the isochronous watermark for the soft frame...

...an isochronous function
is dependent on the queue depth and dequeuing rate of the isochronous **peripheral** . Typically, a margin of reliability is also employed.

Preferably, the latency requirements of the isochronous...

...and the transmit times of isochronous data and
asynchronous transfers are gathered from the interconnected **peripherals** as an integral part of the power on/reset initialization process. Alternatively, the latency requirements and the transmit times may be gathered post initialization with **connection** management transactions.

Upon having set the soft frame **size** , computed the isochronous watermark, and determined the transmit times of the isochronous data and the...

...its latency requirement, and at each polling it is guaranteed to be polled with a **bandwidth** sufficient for its transmit time.

The soft frame used to schedule the isochronous functions is scheduled polling of the isochronous functions, the **bus** controller 14 determines the **number** of asynchronous transfers needed, i.e. the **number** of asynchronous functions to be polled, the **number** of **connection** management transactions to be polled to detect live **attachment** /detachment of serial bus elements, etc., step 310. The bus controller 14 further determines the **amount** of asynchronous transfer time available **per** soft frame, by subtracting the cumulative transmit time of the scheduled isochronous functions from the frame watermark **portion** of a soft frame, step 312. Finally, the **bus** controller 14 schedules the asynchronous transfers in one or more soft frames, around the polling...

...Figure 16.

In some embodiments, the above described process is repeated upon detection of live **attachment** /detachment of serial **bus** elements, thereby **dynamically** adapting in **real time** the frame based polling schedule to the serial bus elements actually present. The manner in...

...performed is slightly different when the process is repeated in response to detection of live **attachment** /detachment of serial bus elements. These differences will be noted when step 308 is described...

...316,
Le. whether it is scheduling at power on/reset or in response to live **attachment** /detachment of serial bus elements. If the scheduling is being performed for the first schedule...being performed to generate the first polling schedule, i.e. it is being performed to **dynamically** adapt the polling schedule, the **bus** controller 14 further determines if the lowest latency is less than or equal to the' "old" soft frame **size** , step 330. If the lowest latency is less than or equal to the "old" soft frame **size** , the bus controller 14 proceeds to step 332, otherwise it proceeds with step 352.

Continuing...
...If the isochronous watermark has been exceeded, the bus controller 14 reestablishes the soft frame **size** and the corresponding isochronous watermark, excluding the isochronous function that yielded the lowest latency, step...

...functions with low latencies earlier at step 338 to yield the eventual "successful" soft frame **size** .

Continuing now with Figure 15c, recall that the bus controller 14 proceeds with step 352...

...decided back at step 330 the lowest latency is greater than the "old" soft frame **size** . Steps 352 and 354 are the same as steps 332 and 334 described earlier. However...

...necessary, since the lowest
latency was determined to be greater than the "old" soft frame **size**
back at step 330. Thus, it is not possible for the bus controller 14 to
...Voice channel, and two
asynchronous functions, i.e. Mouse and Keyboard. Additionally,
asynchronous transfers for **connection** management (GEO poll),
attention requests and management queries also have to be
accommodated. The latencies...

...respectively, step 302a.

Thus, the lowest latency is determined to be 364us; the soft frame **size**
is set to 364us; and the isochronous watermark is computed to be
273us (using an...step 308a.

Next., the bus controller 14 determines that polling for
two asynchronous functions, five **connection** management
transactions (GEO poll), one attention request, and three management
queries are required, step 310...

...of the present invention for generating a frame
based polling schedule that guaranties latencies and **bandwidths** of
isochronous functions. The first three steps, i.e. steps 382 - 386 are
identical to...

...However, step 396 differs from step 312 in that the bus controller 14
determines the **amount** of transmit time available in each base soft
frame of the set of base soft transfers are scheduled using **multiple**
sets of soft frames, instead of
multiple soft frames. Step 398 will be described in more detail below
with additional references to...

...20.

In some embodiments, the above described process is also
repeated upon detection of live **attachment** /detachment of serial **bus**
elements, thereby **dynamically** adapting in **real time** the frame based
polling schedule to the serial bus elements actually present. The
manner in...

...performed is slightly different when the
process is repeated in response to detection of live
attachment /detachment of serial bus elements. These differences will
be noted when step 392 is described...

...i.e. whether it is scheduling at power on/reset or in
response to live **attachment** /detachment of serial bus elements. If the
scheduling is being performed for the first schedule...

...of
a set of soft frames to exceed its isochronous watermark, step 404. The
set **size** is equal to the largest truncated multiplier value of the
isochronous functions. If scheduling would...being performed to generate
the first polling schedule, i.e. it is
being performed to **dynamically** adapt the polling schedule, the **bus**
controller 14 further determines if the lowest latency is less than or
equal to the "old" soft frame **size** , step 412. If the lowest latency is
less
than or equal to the "old" soft frame **size** , the bus controller 14
proceeds to step 414, otherwise it proceeds with step 434.

Continuing...

...one base soft
frame has been exceeded, the bus controller 14 reestablishes the soft frame **size** for the base soft frames and the corresponding isochronous watermarks, excluding the isochronous function that...

...with low latencies earlier at step 420 to yield the eventual
it successful" soft frame **size** for the base soft frames
Continuing now with Figure 19c, recall that the bus
controller...
...decided back at step 412 the
lowest latency is greater than the "old" soft frame **size** . Steps 434 and 436 are the same as steps 414 and 416 described earlier. However...

...necessary, since the lowest
latency was determined to be greater than the "old" soft frame **size** back at step 412. Thus, it is not possible for the bus controller 14 to
...respectively, step
382a. Thus, the lowest latency is determined to be 364us; the soft frame **size** is set to 364us; and the isochronous watermark is computed to be 273us (using an...

...frames, step
392a.

Next, the bus controller 14 determines polling for two
asynchronous functions, five **connection** management transactions
(GEO poll), one attention request, and three management queries are
required, step 394a...

...resulting polling schedule for this exemplary
application is shown in Figure 21b.
Thus, methods and **apparatus** for dynamically generating
and maintaining frame based polling schedules for polling
isochronous and asynchronous functions that guaranty latencies and **bandwidths** to the isochronous functions have been described. For additional information about the method and **apparatus** of the present invention, refer to the enclosed **Appendices** .

While the present invention has been described in terms
of the above embodiments, those skilled...

...will recognize that
the invention is not limited to the embodiments described. The method and **apparatus** of the present invention can be practiced with modification and alteration within the spirit and scope of the **appended** claims. The description is thus to be regarded as illustrative instead of limiting on the...

Claim
1 An **apparatus** for generating and maintaining frame based
polling schedules that guaranty latencies and **bandwidths** to
isochronous functions of isochronous **peripherals** hierarchically
interconnected to a system unit of a computer system through a serial **bus** assembly including a **bus** controller, zero or more **bus** signal **distributors** , and one or more **bus** **interfaces** , said **apparatus** comprising:
first means for determining a **plurality** of critical operating

characteristics of each isochronous **peripheral** , including latency requirement of the isochronous **peripheral** and transmit time of the isochronous **peripheral** 's isochronous data;
second means for generating and maintaining a polling schedule consisting a superframe...

...as frequent as it is necessary based on the determined critical operating characteristics.

2 The **apparatus** as set forth in Claim 1, wherein said first means for determining a **plurality** of critical operating characteristics of each isochronous **peripheral** comprises first circuitry and complementary logic disposed on said bus controller for polling said isochronous **peripherals** to provide latency requirements of isochronous functions and transmit times of their isochronous data.

3 The **apparatus** as set forth in Claim 2, wherein said first circuitry and complementary logic disposed on said bus controller polls said isochronous **peripherals** for said latency requirements of isochronous functions and transmit times of the isochronous functions' isochronous data at power on and reset.

4 The **apparatus** as set forth in Claim 3, wherein said first circuitry and complementary logic disposed on said bus controller further polls said isochronous **peripherals** for said latency requirements of isochronous functions and transmit times of the isochronous functions' isochronous data periodically during operation.

5 The **apparatus** as set forth in Claim 1, wherein said first means is further used for determining a **plurality** of critical operating characteristics of a **plurality** of asynchronous **peripherals** , including transactional requirements and transmit times of the asynchronous **peripherals** ' transactions, said asynchronous **peripherals** being also **coupled** to said system unit hierarchicallv through said serial bus assembly.

6 The **apparatus** as set forth in Claim 5, wherein said first means comprises first circuitry and complementary logic disposed on said bus controller for polling said isochronous **peripherals** to provide latency requirements of isochronous functions and transmit times of their isochronous data, and for polling said asynchronous **peripherals** to provide said transactional requirements and said transmit times of the asynchronous **peripherals** ' transactions.

7 The **apparatus** as set forth in Claim 6, wherein said first circuitry and complementary logic disposed on said bus controller polls said isochronous **peripherals** for said latency requirements of isochronous functions and transmit times of the isochronous functions' isochronous data, as well as said asynchronous **peripherals** for said transactional requirements and said transmit times of the asynchronous **peripherals** ' transactions, at power on and reset.

8 The **apparatus** as set forth in Claim 7, wherein said first circuitry and complementary logic disposed on said bus controller further polls said isochronous **peripherals** for said latency requirements of isochronous functions and transmit times of the isochronous functions' isochronous data, as well as said

asynchronous **peripherals** for said transactional requirements and said transmit times of the asynchronous **peripherals** ' transactions, periodically during operation.

9 The **apparatus** as set forth in Claim 1, wherein said second means for generating and maintaining a...

...a soft frame for polling isochronous functions to ensure the guaranties are met.

10 The **apparatus** as set forth in Claim 9, wherein, said second means for generating and maintaining a...

...isochronous functions that can not be polled within said constrained base soft frame.

11 The **apparatus** as set forth in Claim 10, wherein, said first circuitry and complementary logic is further...

...said second circuitry and complementary logic is further used for scheduling polling of said isochronous **peripherals** having scheduled isochronous functions for **connection** management transactions,. around said scheduled polling of said isochronous functions for data communication transactions, subject...

...circuitry and complementary logic replicating said constrained base soft frame to schedule said polling for **connection** management transactions, if necessary.

12 The **apparatus** as set forth in Claim 11, wherein, said computer system further includes asynchronous **peripherals** **coupled** to said system unit hierarchically through said serial bus assembly;
said polling schedule generated by said second means further includes polling of said asynchronous **peripherals** for asynchronous transactions;
said second circuitrV and complementary logic is further used for scheduling polling of said asynchronous **peripherals** for asynchronous and **connection** management transactions, around said scheduled polling of said isochronous functions for data communication transactions, subject...

...logic similarly replicating said constrained base soft frame to schedule said polling of said asynchronous **peripherals** for asynchronous and **connection** management transactions, if necessary.

13 The **apparatus** as set forth in Claim 9, wherein, said second means for generating and maintaining a...

...on their respective update rates required which are computed from said latency requirements, determining a **plurality** of truncated integer multipliers, one for each isochronous function, based on said least common update...

...update rates required by said isochronous functions, and scheduling polling of said isochronous functions using **multiple** base soft frames and in accordance ... isochronous functions that can not be polled within said

constrained base soft frames.
14 The **apparatus** as set forth in Claim 13, wherein,
said first circuitry and complementary logic is further...

...said second circuitry and complementary logic is further used for scheduling polling of said isochronous **peripherals** having scheduled isochronous functions for **connection** management transactions, around said scheduled polling of said isochronous functions for data communication transactions, subject...

...circuitry and
complementary logic replicating said constrained base soft frames to schedule said polling for **connection** management transactions, if necessary.

15 The **apparatus** as set forth in Claim 14, wherein,
said computer system further includes asynchronous **peripherals** **coupled** to said system unit hierarchically through said serial bus assembly;
said polling schedule generated by said second means further includes polling of said asynchronous **peripherals** for asynchronous transactions;
said second circuitry and complementary logic is further used for scheduling polling of said asynchronous **peripherals** for asynchronous and connection management transactions, around said scheduled polling of said isochronous functions for data communication transactions, subject to said second constraint, said second circuitry and complementary logic **similarly** replicating said constrained base soft frames to schedule said polling of said asynchronous **peripherals** for asynchronous and **connection** management transactions, if necessary.

16 In a computer system comprising a **bus** controller, at least one **bus** signal **distributor** , and a **plurality** of isochronous **peripherals**

having a **plurality** of isochronous functions, **coupled** to each other in an hierarchical manner, a method for generating and maintaining frame based polling schedules that guaranty latencies and **bandwidths** to the isochronous functions of the isochronous **peripherals** , the method comprising the steps of:
a) determining a plurality of critical operating characteristics of each isochronous **peripheral** , including latency requirement of the isochronous **peripheral** and transmit time of the isochronous **peripheral** 's isochronous data; and
b) generating and maintaining a polling schedule consisting a superframe having further comprises a **plurality** of asynchronous **peripherals**
coupled to the at least one **bus** signal **distributor** and **bus** controller in
an hierarchical manner, and step (a) further comprises determining a **plurality** of critical operating characteristics of the asynchronous **peripherals** , including transactional requirements and transmit times of the asynchronous **peripherals** ' transactions.

21 The method as set forth in Claim 20, wherein step (a) is performed...

...performing any kind of polling to ensure
operational reliability; and
scheduling polling of said isochronous **peripherals** having

scheduled isochronous functions for **connection** management transactions, around said scheduled polling of said isochronous functions for data communication transactions, subject...

...than said frame watermark, replicating said constrained base soft frame to schedule said polling for **connection** management transactions, if necessary.

27 The method as set forth in Claim 26, wherein,
said computer system further includes asynchronous **peripherals** **coupled** to said at least one **bus** signal **distributor** and
bus controller in an hierarchical manner;
said polling schedule generated further includes polling of said asynchronous **peripherals** for asynchronous transactions;
step (b) further comprises scheduling polling of said asynchronous **peripherals** for asynchronous and **connection** management transactions, around said scheduled polling of said isochronous functions for data communication transactions, subject...

...second constraint, replicating said constrained base soft frame to schedule said polling of said asynchronous **peripherals** for asynchronous and **connection** management transactions, if necessary.

28 The method as set forth in Claim 24, wherein, step...

...on their respective update rates required
which are computed from said latency requirements, determining a **plurality** of truncated integer multipliers, one for each isochronous function, based on said least common update...
...update
rates required by said isochronous functions, and scheduling polling of said isochronous functions using **multiple** base soft frames and in accordance to said truncated integer ...performing any kind of polling to ensure
operational reliability; and
scheduling polling of said isochronous **peripherals** having scheduled isochronous functions for **connection** management transactions, around said scheduled polling of said isochronous functions for data communication transactions, subject...

...than said frame watermark, replicating said constrained base soft frames to schedule said polling for **connection** management transactions, if necessary.

30 The method as set forth in Claim 29, wherein,
said computer system further includes asynchronous **peripherals** **coupled** to said system unit hierarchically through said serial bus assembly;
said polling schedule generated further includes polling of said asynchronous **peripherals** for asynchronous transactions;
step (b) further comprises scheduling polling of said asynchronous **peripherals** for asynchronous and **connection** management transactions, around said scheduled polling of said isochronous functions for data communication transactions, subject...

...second constraint, replicating said constrained base soft frames to schedule said polling of said asynchronous **peripherals** for asynchronous and **connection** management transactions, if necessary.

US006615285B1

(12) **United States Patent**
Schafranek et al.

(10) **Patent No.: US 6,615,285 B1**
(45) **Date of Patent: Sep. 2, 2003**

(54) **METHOD AND APPARATUS FOR DYNAMICALLY DETERMINING AN ADDRESS UNIQUELY IDENTIFYING A HARDWARE COMPONENT ON A COMMON BUS**

(75) Inventors: **Alex Schafranek**, West Orange, NJ (US); **Keith Elden Strege**, Branchburg Township, Somerset County, NJ (US)

(73) Assignee: **Lucent Technologies Inc.**, Murray Hill, NJ (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/198,098**

(22) Filed: **Nov. 23, 1998**

(51) **Int. Cl.**[7] **G06F 13/14**
(52) **U.S. Cl.** **710/3**; 340/825.07; 345/156; 345/163
(58) **Field of Search** 710/3, 104; 340/825.07; 345/156, 163

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,413,318	A	* 11/1983	Herrington	364/200
4,638,313	A	1/1987	Sherwood, Jr. et al.	
4,773,005	A	* 9/1988	Sullivan	710/9
4,847,834	A	* 7/1989	Bryant	370/449
5,317,693	A	* 5/1994	Cuenod et al.	710/9
5,680,113	A	* 10/1997	Allen et al.	340/825.05
5,708,831	A	* 1/1998	Schon	395/800
5,745,699	A	* 4/1998	Lynn et al.	709/245
5,835,725	A	* 11/1998	Chiang et al.	709/228
5,877,745	A	* 3/1999	Beeteson et al.	345/156
5,898,845	A	* 4/1999	Frantz et al.	710/104
6,009,479	A	* 12/1999	Jeffries	710/8
6,094,053	A	* 7/2000	Harvey	324/434
6,157,527	A	* 12/2000	Spencer et al.	361/64
6,185,638	B1	* 2/2001	Beardsley et al.	710/36
6,195,712	B1	* 2/2001	Pawlowski et al.	710/19

FOREIGN PATENT DOCUMENTS

EP	0 486 818 A1	5/1992

OTHER PUBLICATIONS

L.A. Baxter et al., "System 75: Communications and Control Architecture," AT&T Technical Journal, vol. 64, No. 1, 153–173 (Jan. 1985).

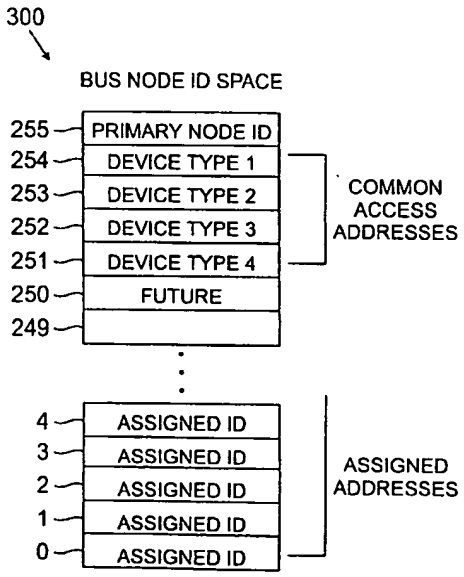
* cited by examiner

Primary Examiner—Abdelmoniem Elamin

(57) **ABSTRACT**

A method and apparatus are disclosed for dynamically determining a bus address that uniquely identifies a hardware component on a common bus. In an illustrative cell station implementation, each cell station includes a hardware controller that communicates on a common bus with a plurality of hardware components. The address space is logically divided into two ranges, namely, a first range of addresses referred to as the common access addresses, and a second range of addresses referred to as the assigned addresses. One address is reserved for the primary node that is responsible for assigning addresses, such as the hardware controller. Thus, each hardware component communicates with the hardware controller on one of the common access addresses to request a channel, and thereafter communicates on an assigned address, once assigned by the hardware controller. Collisions between multiple hardware components on the common access channels are minimized by assigning multiple channels.

14 Claims, 3 Drawing Sheets



39/3,K/78 (Item 78 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

01153614
A method and apparatus **dynamically determining an address uniquely identifying a** hardware component **on a common bus**
Verfahren und Vorrichtung zur dynamischen Bestimmung einer einzigen Adresse zur Indentifizierung einer Komponente in einem gemeinsamen Bus
Procede et dispositif pour allouer dynamiquement une adresse unique pour l'identification d'un composant sur un bus commun
PATENT ASSIGNEE:
LUCENT TECHNOLOGIES INC., (2143720), 600 Mountain Avenue, Murray Hill, New Jersey 07974-0636, (US), (Applicant designated States: all)
INVENTOR:
Schafranek, ALex, 35 Birchwood Avenue, West Orange, New Jersey 07052, (US)
Strege, Keith Elden, 117 Howell Drive, Branchburg Township, New Jersey 08876, (US)
LEGAL REPRESENTATIVE:
Johnston, Kenneth Graham et al (32381), Lucent Technologies (UK) Ltd, 5 Mornington Road, Woodford Green Essex, IG8 OTU, (GB)
PATENT (CC, No, Kind, Date): EP 1004965 A2 000531 (Basic)
EP 1004965 A3 020807
APPLICATION (CC, No, Date): EP 99309078 991115;
PRIORITY (CC, No, Date): US 198098 981123
DESIGNATED STATES: DE; FI; FR; GB; SE
EXTENDED DESIGNATED STATES: AL; LT; LV; MK; RO; SI
INTERNATIONAL PATENT CLASS: **G06F-013/40** ; **G06F-012/02** ; **G06F-012/06**
ABSTRACT WORD COUNT: 150
NOTE:
Figure number on first page: 1

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	200022	564
SPEC A	(English)	200022	2810
Total word count - document A			3374
Total word count - document B			0
Total word count - documents A + B			3374

A method and apparatus **dynamically determining an address uniquely identifying a** hardware component **on a common bus**
INTERNATIONAL PATENT CLASS: **G06F-013/40** ...

... **G06F-012/02** ...
... **G06F-012/06**

...ABSTRACT A2
A method and **apparatus** are disclosed for **dynamically** determining a **bus** address that uniquely identifies a **hardware component** on a common bus. In an illustrative cell station implementation, each cell station includes a **hardware** controller that communicates on a common bus with a **plurality** of **hardware components** . The address space is logically divided into two ranges, namely, a first range of addresses...

...is reserved for the primary node that is responsible for assigning addresses, such as the **hardware** controller. Thus, each **hardware component** communicates with the **hardware** controller on one of the

common access addresses to request a channel, and thereafter communicates on an assigned address, once assigned by the **hardware** controller. Collisions between **multiple hardware components** on the common access channels are minimized by assigning **multiple** channels.

...SPECIFICATION Application

The present invention is related to United States Patent Application entitled "A Method And **Apparatus** For Determining An Address Uniquely Identifying A **Hardware Component** On A Common Bus," (Attorney Docket **Number** Eby 1-22-3-5), filed contemporaneously herewith, assigned to the assignee of the present...

...by reference herein.

Field of the Invention

The present invention relates to a method and **apparatus** for **assigning bus** addresses, and more particularly, to a method and **apparatus** for **dynamically assigning bus** addresses to **hardware components installed** on equipment, such as a cell station.

Background of the Invention

Conventional cellular telephone systems...

...In order for a cell station to perform required functions, the cell station typically incorporates **hardware components** , often referred to as "circuit packs." The same cell station, for example, might include a set of transceivers and other **hardware** equipment, such as control and timing boards, for communicating with cellular telephone units in a known manner. In addition, a cell station can include one or more **interface** boards for communicating, for example, on a T1 line with a switch of the PSTN...

...order for the cell station to properly communicate with and supervise each of the various **hardware components** , the cell station also includes a **hardware** controller. The **hardware** controller and each of the **hardware components** are typically stored by the ceii station on a frame or rack.

In the event of a **hardware component** failure or when routine servicing is required, a given **hardware component** on a cell station might be replaced by a compatible piece of **hardware** , having a different manufacturer, model **number** or version. In addition, additional **hardware components** may be added to the cell station to increase the capacity of the cell station, or the cell station **configuration** may otherwise be modified. As each new **hardware component** is added to a cell station, the **hardware component** must typically be associated with a particular sector (antenna) and carrier frequency on the cell station.

Each **hardware component** typically communicates with the **hardware** controller and with one another on a common bus. In order to differentiate each **hardware component** on the common **bus** , each **hardware component** is **assigned** a unique address. The **hardware** controller maintains a directory that maps the sector and carrier frequency associated with each **hardware component** , as well as the corresponding bus address. The system **configuration** information stored by the **hardware** controller must match the physical **hardware components** that are actually **installed** on the cell station.

Some systems for determining system **configuration** information, however, including sector, carrier frequency and **bus assignments** for each **hardware component** , are labor-intensive and require considerable manual effort. Specifically, an operator typically manually enters the

relationship between the physical address of each **hardware component** and the corresponding logical **connection** (sector and carrier frequency) for all **hardware components installed** on a given cell station. In addition, each slot often has a prewired address. When a **hardware component** is plugged into the slot, the **hardware component** reads the address value from the backplane and uses the address to communicate with other **hardware components** and the **hardware** controller on the common bus. Since the prewired backplane address has little or no physical...

...information is needed to convert the backplane address to an understanding of the type of **hardware component** that is at that address, as well as the sector and carrier frequency associated with the **hardware component** .

As apparent from the above-described deficiencies with conventional techniques for obtaining system configuration information, a need exists for a technique for automatically determining system configuration information and for **assigning** addresses on a common bus.

Summary of the Invention

Generally, according to one aspect of the invention, a method and apparatus are disclosed for **dynamically** determining an address that uniquely identifies a hardware component on a common **bus** . In an illustrative cell station implementation, each cell station includes a hardware controller that communicates on a common **bus** with a **plurality** of **hardware components** .

According to an aspect of the invention, each **hardware component** **dynamically** obtains a **bus** address that uniquely identifies the **hardware component** on the common **bus** . The address space is logically **divided** into two ranges, namely, a first range of addresses referred to as the common access...

...is reserved for the primary node that is responsible for assigning addresses, such as the **hardware** controller. Thus, each **hardware component** communicates with the **hardware** controller on one of the common access addresses to request a channel, and thereafter communicates on an assigned address, once assigned by the **hardware** controller.

Collisions between **multiple hardware components** on the common access channels are minimized by allocating one address in the common access address space for each **device** type. Thus, **hardware components** of a different type can execute the access procedure in parallel since they access on a different common access channel. In an alternate implementation, collisions are minimized by allocating a **plurality** of common access addresses and having each **hardware component** randomly select a common access address. **Assigning multiple** channels for common access reduces the **bus** initialization time, since **hardware components** are less likely to have to wait for the appropriate access channel.

A **hardware component** may access the **bus** to obtain an **assigned** address by first listening to hear if any other **hardware component** is transmitting using the same common access channel. If the common access channel is clear, the **hardware component** sends an identification message to the **hardware** controller using the address reserved for the **hardware** controller. The common access address is associated with the identification message so the **hardware** controller may respond to the **hardware component** with an address assignment message on the **hardware component** 's common access address. The address assignment message carries a new address assignment chosen by the **hardware** controller from the assigned address space. All further communication by the accessing **hardware component** takes place using the assigned address. Thereafter,

the common access address is available for another **hardware component** to use.

A more complete understanding of the present invention, as well as further features...

...in accordance with the present invention;

FIG. 2 is a schematic block diagram illustrating a **hardware component** of FIG. 1, in accordance with the present invention;

FIG. 3 illustrates an illustrative **partitioning** of the **bus** address space in accordance with the present invention to **allocate** common access channels and assigned addressses; and

FIG. 4 is a flow chart describing an exemplary system **configuration** process performed by the **hardware component** of FIG. 2.

Detailed Description

FIG. 1 is a block diagram showing the architecture of...

...station 100 in accordance with the present invention. The cell station 100 preferably includes a **hardware** controller 110 that communicates on a common **bus** 115 with a **plurality** of **hardware components** 200A-200H. The **hardware components** 200A-200H may be **installed** , for example, on one or more frames or racks 130 on the cell station 100. The **hardware components** 200A-200H, such as an illustrative **hardware component** 200, are discussed further below in conjunction with FIG. 2.

According to a feature of the present invention, each **hardware component** 200A-200H **dynamically** obtains a **bus** address that uniquely identifies the **hardware component** 200A-200H on the common **bus** 115. As shown in FIG. 3, the present invention logically **divides** the address space into two ranges, namely, a first range of addresses referred to herein...

...addresses, and a second range of addresses referred to herein as the assigned addresses. The **size** of each address space can change to meet the needs of a given system, as...

...ordinary skill in the art. In theory, the present invention supports up to N-2 **hardware components** , where N is the **size** of the address space (with the two exceptions being the **hardware** controller 110 and at least one common access address). Thus, each **hardware component** communicates with the **hardware** controller 110 on one of the common access addresses to request a channel, and thereafter communicates on one of the **assigned** addresses, once **assigned** by the **hardware** controller 110.

According to a further feature of the invention, collisions between **multiple hardware components** on the common access channels are minimized. In one embodiment, collisions are minimized by allocating one address in the common access address space for each **device** type. Thus, for example, all of the **hardware components** of a particular **device** type would request an address using address 253. Thus, **hardware components** of a different type can execute the access procedure in parallel since they access on a different common access channel. If two **hardware components** of the same type both try to access the common bus 115 at the same...

...handled, as in typical random access schemes, by including a unique value from the accessing **hardware component** in the address assignment message exchange. The unique value could be some **hardware** -derived **number** , such as a serial **number** , or simply a random **number** . In this manner, the accessing **hardware component** can ensure that the address

assignment message is intended for it by checking the address...

...for the appropriate unique value.

In an alternate implementation, collisions are minimized by allocating a **plurality** of common access addresses and having each **hardware component** randomly select a common access address. **Assigning multiple** channels for common access also serves to reduce the **bus** initialization time, since **hardware components** are less likely to have to wait for the appropriate access channel.

Generally, bus initialization begins when the **hardware** controller 110 begins transmitting on the common bus 115. The **hardware components** 200A-200H do not transmit on the common bus 115 until the **hardware** controller 110 first transmits something. Once the **hardware** controller 110 transmits on the common bus 115, then a **hardware component** 200A-200H may access the bus 115 by first listening to hear if any other **hardware component** 200A-200H is transmitting using the same common access channel. If the common access channel is clear, the **hardware component** sends an identification message to the **hardware** controller 110 using the address assigned to the **hardware** controller 110. The common access address is associated with the identification message so the **hardware** controller 110 may respond to the **hardware component** .

The **hardware** controller 110 then sends an address assignment message back to the **hardware component** 200A-200H on the **hardware component** 's common access address. The address assignment message carries a new address assignment chosen by the **hardware** controller 110 from the assigned address space. For example, the address assignments may be assigned sequentially from the assigned address space. All further communication by the accessing **hardware component** takes place using the assigned address. Thereafter, the common access address is available for another **hardware component** to use.

Once all of the **hardware components** have executed the access procedure, they continue to operate using their assigned address until they execute a **hardware** reset, upon which the **hardware component** must go through the access procedure again to obtain a (possibly different) address. In addition, once the **bus** addresses for the **hardware component** are **assigned** in accordance with the present invention, the cell station 100 operates normally, forwarding the messages that are received from the central control station (not shown) to the **hardware components** 200A-200H and vice versa, as required.

As shown in FIG. 1, the cell station 100 preferably includes a **hardware** controller 110 and related memory, such as a data storage **device** 120, which may be distributed or local. The **hardware** controller 110 may be embodied as a single processor, or a **number** of local or distributed processors operating in parallel. The data storage **device** 120 and/or a read only memory (ROM) are operable to store one or more instructions, which the **hardware** controller 110 is operable to retrieve, interpret and execute.

The data storage **device** 120 preferably includes the code 150 to perform supervisory, control functions and other conventional functions ...

...States Patent No. 4,829,554, incorporated by reference above. In addition, the data storage **device** 120 includes a **hardware component configuration** database 160 for storing the carrier frequency, frame, sector **number** (such as (alpha), (beta), (gamma)), unit type and unit **number** associates with each **hardware component** on the cell station 100. In addition, the data storage **device** 120 includes a system **configuration** process 400, discussed below in conjunction with FIG. 4, that communicates with the **hardware components** 200A-200H to compile

the **configuration** information and generate the **hardware component configuration** database 160.

It is noted that the cell station 100 can provide the local **configuration** information to a central cell controller (not shown), as would be apparent to a person...

...skill in the art. In this manner, the central cell controller can remotely determine the **configuration** of each cell station, such as the cell station 100.

FIG. 2 is a block diagram showing the architecture of an illustrative **hardware component** 200 in accordance with the present invention. As shown in FIG. 2, each **hardware component** 200 includes a processor 210 and related memory, such as a data storage **device** 220, which may be distributed or local. The processor 210 may be embodied as a single processor, or a **number** of local or distributed processors operating in parallel. The data storage **device** 220 and/or a read only memory (ROM) are operable to store one or more instructions, which the processor 210 is operable to retrieve, interpret and execute.

The data storage **device** 220 preferably includes the code 250 to perform conventional functions for communicating with the **hardware** controller 110 of FIG 1. In addition, the data storage **device** 120 includes a system **configuration** process 400, discussed below in conjunction with FIG. 4, that cooperates with the **hardware** controller 110 to **dynamically** obtain a **bus** address for the **hardware component** and to provide information identifying the **hardware component** to the **hardware** controller 110.

As previously indicated, the **hardware** controller 110 and each **hardware component** 200 cooperatively implement a system **configuration** process 400, such as the illustrative process shown in FIG. 4, to **dynamically assign bus** addresses for each **hardware component** and to provide information identifying the **hardware component** to the **hardware** controller 110.

As shown in FIG. 4, the system **configuration** process 400 is initiated during step 410 upon the powering up of the **hardware component** 200. Thereafter, the **hardware component** listens for other **hardware components** during step 420 on a common access channel. In the illustrative embodiment, the **hardware component** listens for other **hardware components** on a common access channel assigned for its **device** type.

A test is performed during step 430 to determine if the common access channel...

...determined during step 430 that the common access channel is not being used, then the **hardware component** sends a message to the **hardware** controller 110 on the appropriate common access channel during step 440, and listens for a response on the appropriate common access channel. The initial message sent to the **hardware** controller 110 may include, for example, an indication of the **configuration** information for the **hardware component** , such as the carrier frequency, sector, unit type and unit **number** .

The **hardware** controller 110 then sends an address assignment to the **hardware component** during step 450 using the common access channel upon which the request was received. As previously indicated, the address assignment may include a unique identifier of the **hardware component** .

A test is performed by the **hardware component** during step 460 to determine if the address assignment is for this **hardware component** . If it is determined during step 460 that the address assignment is not for this **hardware component** , then the **hardware component** waits

for the next address assignment message (or may retransmit its request) during step 470...

...If, however, it is determined during step 460 that the address assignment is for this **hardware component** , then the **hardware component** begins using the assigned address for further communications during step 480.

It is to be...

CLAIMS 1. A method for **dynamically** determining an address that uniquely identifies a **hardware component** on a common bus, said method comprising the steps of:

requesting an address on a common access channel; and

receiving an **assigned** address for further communication on said **bus** .

2. The method **according** to claim 1, further comprising the step of **assigning** a **plurality** of said common access channels.
3. The method **according** to claim 2, further comprising the step of **assigning** at least one common access channel for each type of said **hardware component** .
4. The method **according** to claim 2, wherein said address request is sent on a randomly selected one of said **plurality** of common access channels.
5. The method according to claim 1, further comprising the step of transmitting with said address request identification information describing said **hardware component** and how said **hardware component** is located within a larger piece of equipment.
6. A method for **allocating** an address space on a common **bus** utilized by a **plurality** of **hardware components** , said method comprising the steps of:

allocating at least one address in said address space as a common access channel for requesting addresses; and

allocating a range of addresses in said address space as assigned addresses for communicating with said...

...channel; and

receive an assigned address for further communication on said bus.

11. The system **according** to claim 10, wherein said processor is further **configured** to **assign** a **plurality** of said common access channels.
12. The system **according** to claim 11, wherein said processor is further **configured** to **assign** at least one common access channel for each type of said hardware component.
13. The system **according** to claim 11, wherein said address request is sent on a randomly selected one of said **plurality** of common access channels.
14. The system **according** to claim 10, wherein said processor is further **configured** to transmit with said address request identification information describing said hardware component and how said **hardware component** is located within a larger piece of equipment.
15. A system for **allocating** an address space on a common bus utilized by a plurality of **hardware components** , said system comprising:

a memory for storing computer readable code; and

a processor, operatively **coupled** to said memory, said processor **configured** to:

allocate at least one address in said address space as a common access channel for requesting addresses; and

allocate a range of addresses in said address space as **assigned**

addressee for communicating with said **hardware components** .

16. The system **according** to claim 15, wherein a **plurality** of addresses **allocated** to said common access channels.
17. The system **according** to claim 16, wherein said processor is further **configured** to **assign** at least one common access channel for each type of said **hardware component** .
18. The system according to claim 16, wherein a **hardware component** randomly selects one of said common access channels for requesting an address

US005550990A

United States Patent [19]

Keener et al.

[11] Patent Number: 5,550,990

[45] Date of Patent: Aug. 27, 1996

[54] **PHYSICAL PARTITIONING OF LOGICALLY CONTINUOUS BUS**

[75] Inventors: **Don S. Keener**, Boca Raton; **Andrew B. McNeill**, Deerfield Beach; **Thomas H. Newsom**, Boca Raton; **Kevin L. Scheiern**, Deerfield Beach; **Richard W. Voorhees**; **Edward I. Wachtel**, both of Boca Raton, all of Fla.

[73] Assignee: **International Business Machines Corporation**, Armonk, N.Y.

[21] Appl. No.: **400,425**

[22] Filed: **Mar. 3, 1995**

Related U.S. Application Data

[63] Continuation of Ser. No. 901,337, Jun. 19, 1992, abandoned.

[51] **Int. Cl.**[6] **G06F 13/40**

[52] **U.S. Cl.** **395/309**; 395/308

[58] **Field of Search** 395/325, 275, 395/250, 725, 425, 309, 306, 308

[56] **References Cited**

U.S. PATENT DOCUMENTS

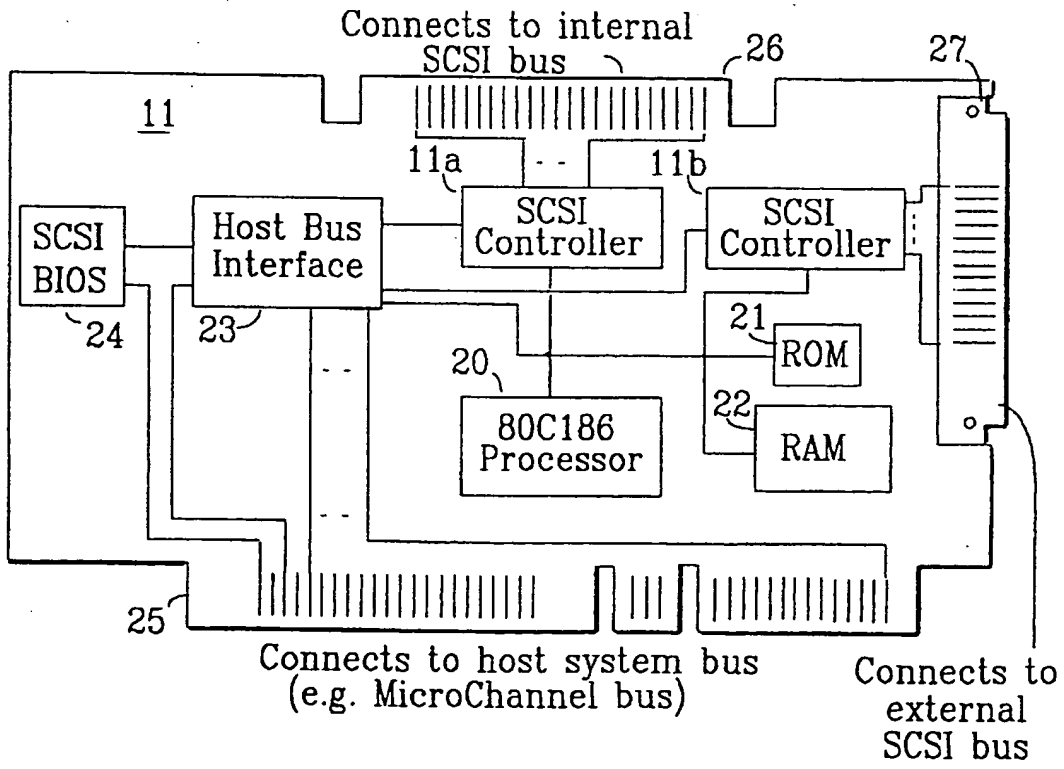
4,716,525	12/1987	Gilanyi et al.	395/250
4,975,829	12/1990	Clarey et al.	395/275
5,117,486	5/1992	Clark et al.	395/250
5,140,691	8/1992	Austruy et al.	395/575
5,191,653	3/1993	Banks et al.	395/275
5,191,656	3/1993	Forde, III et al.	395/325
5,274,783	12/1993	House et al.	395/325

Primary Examiner—Jack B. Harvey
Assistant Examiner—Glenn A. Auve

[57] **ABSTRACT**

Arrangements for physically partitioning a bus having a well defined architecture as a physical entity, wherein the partitioning is logically transparent to a computer and devices which communicate through the bus and serves to avoid problems potentially arising because of the scope of actions permitted by the architecture. A typical bus architecture to which present arrangements have relevance is that associated with SCSI (Small Computer System Interface) buses. The potential problems allowed to occur architecturally involve: (a) exposures of data security/integrity; (b) excessive signal degradation due to use of signal rates which although allowed by the architecture are inappropriate for a particular bus loading environment also allowed by the architecture; (c) restrictions preventing parallel transfer of data between the computer and multiple storage devices; (d) restrictions unduly limiting the number of devices attachable to one logical bus path (one input-output channel of the computer). The disclosed arrangement partitions the bus into two or more physical entities which to the computer appears as one logical entity.

6 Claims, 6 Drawing Sheets



39/3,K/115 (Item 115 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00573301
Physical partitioning **of logically** continuous bus .
Physische Aufteilung eines logisch kontinuierlichen Bus.
Partage physique de bus logiquement continu.
PATENT ASSIGNEE:
INTERNATIONAL BUSINESS MACHINES CORPORATION, (200125), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB;IT)
INVENTOR:
Keener, Don Steven, 1019 West Royal Palm Road, Boca Raton, Florida 33486, (US)
McNeill, Andrew Boyce, 181 NW 41 Way, Deerfield Beach, Florida 33442, (US)
Newsom, Thomas Harold, 753 Camino Lakes Circle, Boca Raton, Florida 33486 , (US)
Scheiern, Kevin Lee, 2832 Waterford Drive South, Deerfield Beach, Florida 33442, (US)
Voorhees, Richard W., 4072 NW 3rd Avenue, Boca Raton, Florida 33431, (US)
Wachtel, Edward Irving, 22167 Serenata Circle East, Boca Raton, Florida 33433, (US)
LEGAL REPRESENTATIVE:
Burt, Roger James, Dr. (52152), IBM United Kingdom Limited Intellectual Property Department Hursley Park, Winchester Hampshire SO21 2JN, (GB)
PATENT (CC, No, Kind, Date): EP 575042 A1 931222 (Basic)
APPLICATION (CC, No, Date): EP 93303471 930504;
PRIORITY (CC, No, Date): US 901337 920619
DESIGNATED STATES: DE; FR; GB; IT
INTERNATIONAL PATENT CLASS: **G06F-013/40**
ABSTRACT WORD COUNT: 174

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPABF1	447
SPEC A	(English)	EPABF1	5179
Total word count - document A			5626
Total word count - document B			0
Total word count - documents A + B			5626

Physical partitioning **of logically** continuous bus .
INTERNATIONAL PATENT CLASS: **G06F-013/40**

...ABSTRACT A1
Arrangements are disclosed for physically **partitioning** a **bus** having a well defined architecture as a physical entity, wherein the partitioning is logically transparent to a computer and **devices** which communicate through the bus and serves to avoid problems potentially arising because of the...

...architecture to which present arrangements have relevance is that associated with SCSI (Small Computer System **Interface**) buses. The potential problems allowed to occur architecturally involve: (a) exposures of data security/integrity...

...allowed by the architecture; (c) restrictions preventing parallel transfer of data between the computer and **multiple** storage **devices** ; (d) restrictions unduly limiting the **number** of **devices** **attachable**

to one logical **bus** path (one input-output channel of the computer). The disclosed arrangement **partitions** the **bus** into two or more physical entities which to the computer appears as one logical entity...

...SPECIFICATION Of the Invention

This invention relates to bus systems for transferring data between computers and **peripheral devices** .

Background Of The Invention

Contemporary computer- **peripheral** bus systems, exemplified by the SCSI (Small Computer System **Interface**) bus as specified by American National Standards Institute (ANSI)
(Footnote: (sup 1) Refer to Document...

...2/86109, both published and copyrighted by ANSI)
(Footnote: (sup 1))
, allow for a variable **number** of **devices** to be positioned at variable distances from each other and from a host computer, along an electrically **continuous bus** , and to exchange data signals with the computer in baseband pulse form. Many such systems...

...signals transmitted may have sharp edge transitions. It has been observed that in some system **configurations** these signals may encounter bus impedance conditions preventing accurate reception, which in turn could result in system failures of potentially indeterminate nature.

Also, security of data stored in **devices coupled** to the bus taps, and intended for use of a particular computer **linked** to the bus, could be easily breached by other **devices linked** to the bus.

Furthermore, as such buses evolve to serve **configurations** not contemplated by developers of the original buses, control software executed by the host computer for initializing the bus **configuration** may become useless relative to new **configurations** , adding an undesirable software cost to the process of upgrading to the new **configurations** .

The present invention seeks to provide a bus arrangement which is suitable for efficiently alleviating these problems.

Objects Of The Invention

An object of this invention is to provide a **bus** system, for reliably **linking** varied **numbers** of **peripheral devices** to a computer, wherein **bus connections** of the **devices** may be variably spaced relative to each other and the computer and wherein the bus is logically **configured** to avoid one or more of the problems stated above while presenting to the computer the logical appearance of a physically **continuous bus** .

A data **bus** may be provided for **connecting peripheral devices** to a computer, which is arranged logically to appear as a **continuous bus** entity to the computer, but consists physically of **plural bus partitions** that are isolated from each other and which are accessible to the computer through distinctly...

...rates that can be effectively optimized in relation to transmission distances between the computer and **devices linked** to these paths and thereby effectively optimized for worst case impedance conditions that could be encountered in each **path** .

A **data bus** system for reliably **linking** a computer with **multiple devices** positioned at various signalling distances from the computer and each other, may be arranged wherein data is transferrable between the computer and **devices** at different rates designed to accomplish the transfers efficiently, in a manner transparent to the computer and **devices** , while minimizing the risk of signals representing the data being distorted beyond recognition.

A data **bus** system for reliably **linking** **multiple** data handling **devices** with each other, said **devices** including a computer and at least one storage **device** , may include a **bus** physically **partitioned** so as to selectively prevent **devices** other than the computer from accessing data stored in the storage **device** (s), while maintaining the appearance of the bus relative to the computer as a logically **continuous** entity.

A logically **continuous** but physically **partitioned** data **bus** system, may reliably **link** a computer with **multiple** **devices** , wherein the initial **configuration** of the **bus** can be electrically established under control of the computer, when the computer is running under control software designed specifically to **configure** a physically **continuous** **bus** system.

A data **bus** system for **connecting** a computer with one or more **peripheral** **devices** including at least one storage **device** , may be adaptable to prevent access to the at least one storage **device** by any other **peripheral** **device** which could adversely affect data stored in the at least one storage **device** that is used by the computer. A data bus system, for **connecting** a computer with, **peripheral** devices, wherein the **bus** is made to appear to the computer as a single logical **connection** **path** for **data** , but is physically **partitionable** into plural **segments** that are electrically isolatable from each other can be used to transfer data in distinctly different modes. A **segmentable** bus may be provided as stated previously, wherein data is transferrable concurrently relative to devices...

...A data bus adaptation system may interface to a bus which is physically partitionable into **plural** segments, while maintaining appearance of a logically continuous bus relative to a computer and peripheral devices **attached** to the segments; wherein the adaptation system is operable to maintain **compatibility** with control software in the computer that is designed specifically for **configuring** a **bus** system that is physically continuous.

Summary Of The Invention

According to the invention there is provided a partitionable **bus** adapting unit for **interfacing** between a computer system **bus** and a peripheral **bus** that links to peripheral devices. The adapting unit may be provided with two or more ports which connect separately to discrete **sections** of the **peripheral** bus, relative to a **bus** system in which such **sections** normally would be **continuously** **linked** to each other through the adapting unit. Transfers of data between each unit port and the computer system bus are controlled separately within the adapting unit, while allowing the **peripheral** bus to be **configured** relative to the computer system as if it were logically a single **continuous** **bus** . The adapting unit is thereby capable of providing customized handling of data transfers relative to each of its **peripheral** **bus** ports.

One application of this **customized** handling capability is to transfer data at different rates between the adapting unit ports and different **sections** of the **bus** , so as to avoid potential noise problems relative to one or more of the **bus** **sections** . In this application, the computer system, the adapting unit, and certain "internal" **devices** **linked** to one **peripheral** port of the adapting unit, are all contained within a common housing enclosure; whereas other "external" **devices** that are **linked** to one or more other **peripheral** ports of the adapting unit are located either outside of or inside the common housing. Distances between the external **devices** and the adapting unit are large in relation to distances between the internal **devices** and that unit. Accordingly, data is transferrable between the internal **devices** and the adapting unit at a higher rate than data that is transmitted between the external **devices**

and that unit.

In this and other applications the adapting unit may be adapted to conduct data transfer operations relative to its **peripheral** ports concurrently so that the overall throughput of data relative to all **devices** can be sustained at a higher rate than it could be if the **peripheral bus** were a single physically **continuous** entity.

In another application of this **partitioned** port arrangement, storage **devices** intended for exclusive use and control by the host computer system, but which would be directly accessible to other **devices** following normal practices of the **peripheral** bus architectural standard, are isolated from the external **devices** by having the internal and external **devices coupled** to different **peripheral** ports of the adapting unit.

When applied to the above-mentioned SCSI bus, the housing...

...its power supply can be used to house and power the adapting unit, several "internal" **devices** and **segments** of the **peripheral** /SCSI **bus linking** those **devices** to the adapting unit. Other "external" **devices linked** to the **peripheral** bus are located outside of this housing. In this environment, the spacing between the adapting unit and the internal **devices** is tightly controllable by the designers of the computer system, whereas the spacing between the same unit and the external **devices** in general is left for determination by system users and is therefore not determinable by those designers. Thus, the internal and external **devices** are **configurable** to present impedance conditions relative to the adapting unit which are respectively controllable by and beyond the control of the computer system designers. **Accordingly** , by **partitioning** the adapting unit to serve the internal and external **devices** separately, the rate of data transfer relative to the internal **devices** can be optimized without concern for distortions associatable with external impedances of the same SCSI...

...unpartitioned" SCSI bus arrangement for a computer system.

Figure 2 is a schematic of a **partitioned** SCSI **bus** arrangement in accordance with the invention, for use in the system of Figure 1.

Figures...

...O (input/output) adapter unit 5, shown as a card that plugs into the motherboard, **links** the system to some of its **peripheral devices** , shown at 6 and 7, through a physically **continuous** SCSI (Small Computer System **Interface**) **bus** cable shown at 9. **Devices** 6 and 7 are respectively **designated** as internal and external **devices** ; the internal **devices** being enclosed within housing 2 (and potentially powered with the CPU and other system elements enclosed by the housing), and the external **devices** being located physically outside of the housing. **Devices** 6 are **connected** to internal **section** 9a of **bus** 9, and **devices** 7 are **connected** to external **section** 9b of the same **bus** . **Bus sections** 9a and 9b are interconnected as shown.

Other parts of the system, which are mounted upon or **connected** to the motherboard 3 but are not shown to simplify the illustration, include system memory (random access and read only), timing controls, a system I/O bus which **links** the system to its **peripheral device** adapters including adapter 5, and possibly additional internal **devices** besides those shown at 6 (e.g. disk drive and diskette drive storage **devices** which are not **linked** to SCSI bus 9). A typical "host" computer for such a system could be an...

...and PS/2 are Trademarks of the International Business Machines Corporation)

(Footnote: (sup 2))
, and **devices** 6 and 7 typically may include disk drive storage **devices** , printing **devices** , etc., having "intelligent controls" **configured** in accordance with the SCSI standards. As indicated in the drawing, the system and bus 9 are capable of supporting a combination of up to 7 internal and external **devices** (all **linked** to daisy-chained **segments** of **bus** 9).

Various problems arising from the **continuous** nature of the **bus** 9 are explained next.

1A. Signal Distortion Problem

It should be understood, in general, that internal **devices** 6 would be much closer to each other and to adapter 5 than external **devices** 7 are to each other and the adapter. Furthermore, it should be understood that factors affecting the impedance and signal distorting properties on the internal **section** 9a of bus 9 (lengths of **bus** **segments** between **devices** , **couplings** to individual **devices** , etc.) are, in general, much more controllable by designers of internal parts of system 1 than **comparable** factors affecting impedances of the external **section** 9b.

Furthermore, it is noted that design parameters of the **devices** 6 and 7 which may affect the integrity of data communicated over the bus 9...

...to control; for instance, it may not be possible to ensure that any of such **devices** have internal logic which could compensate for errors due to signal distortion. Also, in systems...

...rate at which data signals can be transferred over and intelligibly received from the external **section** 9b; or, as a corollary, that the unified **bus** arrangement as shown in Figure 1 may not allow for optimal usage of internal **devices** 6.

1B. Data Security/Integrity Problem

In the foregoing environment, security of data stored on internal disk drives that are **connected** to **bus** **section** 9a may be compromised.

The SCSI architecture allows for two-way communication between **devices** **attached** to the bus. **Accordingly** , data stored on internal disk drives, which is intended to be used primarily by CPU...

...and associated internal processing elements of system 1, may become subject to modification by external **devices** in a manner not contemplated by the designers of the internal system **configuration** .

1C. Problem of Bus Width and Load Restrictions

Early versions of the bus architecture define...

...leads, for parallel transfer of 8 bits of data at a time, and a defined **number** of control leads for controlling such transfers. Relative to internal **devices** , and the internal **bus** **segments** , which have a form and spacing determinable by the computer system designer, it may be...

...g. 16 or 32 bits at a time).

Similarly, it may be seen that the **device** load limitations suggested in Figure 1, of no more than seven **device** loads on the entire bus 9 may be unduly restrictive.

1D. Problem of Optimal Bus...

...bus arrangement of Figure 1 is that data transfers between the host computer and the **bus** **sections** are restricted by the **bus** **configuration** . Thus, it is difficult to achieve optimum concurrency of data transfers over both the internal and external **bus** **sections** .

2. Present Invention

As shown in Figure 2, the present invention addresses these problems by **partitioning** the **bus** and **bus** adapter unit. Adapter unit 11 contains

controller **sections** 11a and 11b which **interface** to respective **bus sections** 12a and 12b, to provide the present functions. Although only a two-way partition is...

...use of adapter sections 11a and 11b to sustain different rates of data transfer on **bus sections** 12a and 12b (10 Megahertz on internal **section** 12a and 5 Megahertz on external section 12b), so as to avoid excessive signal distortion on the external section, while permitting optimal usage of internal **devices** 6.

Figure 4 shows how the adapter partitions can be used to restrict access to internal disk storage **device** 6a to the host CPU exclusively.

Figure 5 shows how the partitions can be operated to increase the **number** of **devices** that can be served by the adapter (to a maximum of 30 **devices** , compared with the maximum of 7 indicated in figure 1).

Figure 6 shows how the partitions can be used to allow for concurrent transfers over the internal and external **bus segments** .

Other uses will be described later with reference to Figures 9-11.

Figure 7 illustrates...

...a not-shown socket on the host-system motherboard, it should be understood that the **components** of such a card (integrated circuit chips and **connective** members), could be integrated directly onto the motherboard.

In addition to **bus** controller **sections** 11a and 11b, card 11 contains a microprocessor 20, read only memory (ROM) 21, random access writable memory (RAM) 22, host bus **interface** unit 23, another ROM memory unit 24 for storing BIOS (Basic Input Output System) control information, and **connector** extensions 25-27. **Connector** 25 **links** to the host system and **connectors** 26 and 27 respectively **couple** to internal and external **sections** of the SCSI **bus** (12a, 12b).

Microprocessor 20, which may consist primarily of an Intel 80C186 processor **module** as shown, directs logical operations of units 11a, 11b, 23 and 24, under control of...

...cooperation with units 11a, 11b, and 24, directs the flow of data signals between host **connection interface** 25 and **peripheral connection interfaces** 26 and 27. Data en **route** between host **interface** 25 and internal SCSI **bus interface** 26 is handled through units 23 and 11a, and data en **route** between **interface** 25 and external SCSI **bus interface** 27 is handled through units 23 and 11b. Unit 23 has not-shown buffers for storing data in transit relative to **interface** 25, and units 11a and 11b have not-shown buffers for storing data en **route** between unit 23 and respective SCSI **interfaces** 26 and 27.

Connector extension 25 plugs into a socket on the host system motherboard, and **couples** via that socket to an I/O bus (e.g. to a Micro Channel
(Footnote...

...International Business Machines Corporation)
(Footnote: (sup 3))
bus in a PS/2 host system) which **links** the host system to **peripheral device** adapter units including card 11. **Connector** extensions 26 and 27 pluggably **attach** to ends of SCSI cable segments.

As suggested in Figure 8, the microprogram directing operations of card 11 (through microprocessor 20) may be organized in a hierarchy of **modules** 30-33; including a task supervisor **module** 30, a command processor **module** 31, a data transfer control **module** 32, and a SCSI **interface** handler **module** 33. The microprogram may also contain diagnostic elements not relevant to the present invention, as...

...processor 31 directs execution of commands defining I/O operations performed between the host system **interface** and the **devices linked** to the SCSI bus(es). Command processor 31 directly controls operations of host bus **interface** unit 23 (Fig. 7) and interacts with **modules** 32 and 33 to direct operations of controller units 11a and 11b (Fig. 7).

Data...

...controls data transfer operations of controllers 11a and 11b relative to the SCSI bus. SCSI **interface** handler **module** 33 attends to control signalling functions between controllers 11a and 11b and **devices linked** to the SCSI bus; including the handling of interruption and other requests received at the SCSI bus **interfaces** .

Presently relevant operations of these **modules** are described next with reference to flow diagrams in Figures 9, 9A, 9B, 10 and 11. Figures 9, 9A and 9B together illustrate command processing and **device configuring** operations directed by **module** 31. Figure 10 illustrates **device** initialization operations directed by **module** 32. Figure 11 illustrates interrupt handling functions directed by **module** 33.

Referring to Figure 9, I/O commands are executed relative to individual logical **devices** (LDn). Command execution begins at 40, and branches to one of two sequence paths at...

...the host system). As shown at 42, one Assign command is executed relative to each **device** served by the subsystem. As shown at 43, in the Assign sequence the command processor updates an Assignment Table (in RAM 22, Fig. 7) with information about the respective **device** (obtained by prior actions evoked by the **interface** handler 33). As shown at 44, if operation 43 is completed successfully, a Flag associated with the respective **device** is cleared in order to indicate that the **device** has been initialized and has not yet received a command (to transfer data, etc.). As...

...are completed, control is returned e.g. to task supervisor 30.

Information entered into the **Assignment** Table in operation 43 typically includes the SCSI address (ID) of the **peripheral device** , the logical **device number** LDn **assigned** to the **device** , and the **bus section** (internal/external) to which the **device** is physically **connected** .

When executing a command other than an Assign command, command processor 31 takes sequence path...

...to operating sequence 47, details of which are shown in Figures 9A and 9B. Sequence **connections** between lines extending through these figures are indicated explicitly by **numbers** .

Referring to Figure 9A, the sequence associated with other commands begins at 50 with a...

...The command processor determines the operating mode of the host system by inspecting settings in **configuration** registers on the card (e.g. in unit 23) that are initialized by the host system.

In compatibility mode, the host system is controlled by (operating system and **configuration**) software which views the adapter 11 and its **devices** effectively as if they are situated along a single continuous (unpartitioned) path. This means that...

...support to effectively allow the host to maintain this unified view of the the SCSI **bus sections** . It also means that in this mode the card subsystem can only support operations relative to the maximum **number** of **devices** allowed on a unified bus (i.e. 7 **devices** in the arrangement

of Figure 1).

When the host system is not in **compatibility** mode (also termed "non-**compatibility** " mode herein), it views the card 11 and its attached **peripherals** in their true **partitioned** configuration (and therefore it and the card may also support more **peripherals** ; e.g. 30 as suggested in Figure 5).

Accordingly , when the host system is in **compatibility** mode, the card subsystem must maintain an **Assignment** Table in which its physical **device** addresses may differ from the logical addresses used by the host, and in which the physical **device** addresses indicate the true physical positions of respective **devices** (e.g. external/internal) and their true data transfer modes (e.g. fast/slow).

The...

...In effect, the actions and operations performed for compatibility mode serve to "retrofit" the partitioned **configuration** of the card in a manner which is transparent to the host system. Thus, for...internal).

In performing operation 51, after determining at 50 that the host is in non- **compatibility** mode, the card logic assumes that the local **Assignment** Table, as currently set, indicates the actual physical location of the logical **device** (LDn) to which the command is directed, regardless of the state of the associated **device** Flag (see operation 44, Fig. 9), and uses the location indicated in that table to have command information sent to the associated **device** via the appropriate **bus section** . The assumption indicated in the preceding sentence is reasonable because the local Assignment Table on the card is set initially in accordance with **Assign** commands originated by the host system, and should retain that setting since it **conforms** to the true physical **configuration** of the subsystem. In operation 52, the flag **assigned** to the associated **device** is set so as to indicate that this **device** has been initialized and received at least one command.

For **compatibility** mode, the command processor sequence branches at 54 on the state of the flag associated with the targeted **device** LDn. If that flag is in a set state (indicating that at least one command has been sent to this **device** since card initialization), the Y (yes) branch is taken at 54 to operation 51 discussed earlier, so that the command is sent to the **device** via the path currently specified in the local Assignment Table, and continued via seqúence path 53 with actions and operations shown in Fig. 9b.

If the flag of target **device** Ldn is in a cleared state when examined at 54, the N (no) path is taken to determination 55 which distinguishes the **device** 's location (External or not External) as currently listed in the Assignment Table. If the **device** is listed currently as external (i.e. **linked** to external **bus section** 12b), operation 51 is performed to send the command via the external **bus section** to the physical **device** address currently indicated in the **Assignment** Table. If the **device** is not explicitly **assigned** to the external **bus section** , operation 56 is evoked to attempt to have the command sent to the **device** via the internal **section** of the **bus** (this is characterized as an "attempt" because at this stage of the sequence the **device** targeted by the host may be **connected** to either the external or internal **bus segment**), and the sequence branches at 57 on the results of that attempt.

If the attempt is successful (**device** responded via internal bus), operation 52 is performed to set the flag of the target/responding **device** and continue the command sequence associated with that action (Figure 9B). If the attempt was...

...taken to another phase of the command sequence shown in Figure 9B via

line 58 **connecting** to that Figure.

Referring to Figure 9B, the continuation of the command sequence from point...

...and either return function 61 or further operations 64 and 52 (in Figure 9A, via **linking** path 65).

Operation 60 continues the command process initiated at 51 in Figure 9A through...

...62, which is evoked after an unsuccessful attempt to send the command to the target **device** via the internal bus (via operation 56 in, figure 9A) based on the current information in the local **Assignment** Table, an attempt is made to send the command via the external bus, and the sequence branches **according** to the result of this action.

If attempt 62 is successful (response received from **device**), the **Assignment** Table entry relative to the target **device** is modified via operation 64, to indicate its **linkage** to the external bus, and the sequence continues via line 65 to operation 52 in Figure 9A, thereby setting the flag for the respective **device** and **linking** to "normal" continuing operations 60 in Figure 9B. If attempt 62 is unsuccessful an error is posted (since the **device** is not contactable on either **bus section**) and the operation ends at 61.

Figure 10 illustrates relevant parts of initial operations that...

...card subsystem to cause "rate negotiation" signals to be exchanged between the subsystem and selected **devices** for enabling the subsystem to determine if it should pace data transfers between it and the **device** at a nominal "low" rate, associated with the external **section** of the SCSI **bus** , or at a higher rate associated with all or part of the internal **section** of the **bus** . In general, the selected **devices** are **devices linked** to **portions** of the SCSI **bus sections** on which transfer of data at the higher rate is considered "safe" (unlikely to have signals distorted beyond recognition), taking into account the **number** of **devices** currently **attached** to that **section** , etc. These negotiations may be initiated either by the card, during its initialization process, or by **devices** in the select "category" at any time.

Starting at 70, the subsystem determines at 71 if a given **device** is in the select category permitting it to operate at either the nominal low rate or the higher rate. If a **device** is not in that category the operation concludes via return 72. If the **device** can negotiate, the negotiation is carried out beginning with determination at 73 of whether or not the negotiation has been completed. Determination 73 is required in order to ensure that **multiple** negotiations are not performed relative to the **device** . If determination 73 indicates prior completion of the negotiation, through this or another sequence patha determination is made of which **bus section couples** to the **device** . If the **device** is external (determination Y at 74), determination 75 is made, but if the **device** is internal operation 76 is evoked to complete negotiation of the higher rate, and the...

...75 to ascertain if the higher rate can be used safely relative to this (external) **device** (based on the **device** type and the current state of loading of the external bus, as indicated in the **configuration** registers set by the host system). If the higher rate can be used, negotiation of...

...operation 77, and the sequence concludes via return 72.

Figure 11 characterizes the handling (by **module** 33, Figure 8) of

device interruption requests (received by adapter controller **sections** 11a, 11b at their **interfaces** to SCSI **bus sections** 12a, 12b). As indicated at 80, handler 33 selects these requests using an alternating preferential sequence which alternately favors requests from internal and external **devices** . By this we mean, that if the last order of preference was internal first and...

...the subsystem is selected via action 81 or 82, for transferring data between the requesting **device** and the host system via unit 23, and an associated data transfer process 83 is conducted relative to that path. Thus, if the request is from an internal **device** , a path is selected at 81 through controller 11a and unit 23, and data is transferred between the **device** and an assigned address in host system memory via that path. As noted previously, data...

...units 11a and 23. On the other hand, if the request is from an external **device** , a path is selected at 82 between the **device** and an assigned address in host system memory through (not-shown FIFO buffers in) units ...

...CLAIMS A1

1. A bus adapting system, for **connecting** a computer to a bus that **attaches** to **peripheral devices** in varied **configurations**, that may potentially interfere with the integrity of operation of the computer, comprising:
   means for effectively **partitioning** said **bus** into **segments** which are isolatable from the computer to differing extents, so that operations of the computer cannot be compromised by **devices** that operate in a manner incompatible or inconsistent with the computer or the current **configuration** of the bus.
2. A bus adapting system in accordance with claim 1 wherein said **partitioning** means includes:
   means for **partitioning** said **bus** into physically separate first and second **segments** ; the first **segment** having a predetermined length and the second segment having a variable length; and
   means for...

...the second rate.

3. A bus adapting system in accordance with claim 1 wherein said **partitioning** means includes:
   means for **partitioning** said **bus** into separate first and second **segments** characterized in that **devices** attached to each segment are not directly accessible for communication with **devices** attached to the other **segment** .
4. A **bus** adapting system in accordance with claim 2 wherein each of said **bus segments** is **connectable** to **plural peripheral devices** within a predefined limiting **number** of **devices** .
5. A **bus** adapting system in accordance with claim 2 wherein said computer is located within a housing enclosure having sockets reserved for **attachment** of some of said **peripheral devices** and wherein:
   said first **bus segment** and said adapting system are contained within said enclosure and said second **bus segment** is located outside said enclosure.
6. The **bus** adapting system of claim 5 comprising adapter means **interfacing** between said computer and said first and second **bus segments** , said adapter means being adapted to make said first and second **bus segments** appear to said computer as a single logically

continuous **bus** entity although said **segments** are in fact physically and electrically separated.

7. A **bus** adapting system in accordance with claim 3 having "compatible" and "non-compatible" modes of operation...

...said compatible mode requires said computer to restrict its assignment of logical identities to said **devices** in a manner ensuring that said **devices** are uniquely identifiable regardless of their physical locations relative to said **bus** **segments** , and said non- **compatible** mode does not require such restriction of identity **assignments** ; and wherein said adapter means includes:

means for cooperating with said computer in said **compatible** mode to allow for operations of said computer to be directed to **devices** linked to said first and second **segments** , while the computer is directing such operations to **devices** that are effectively assumed by it to be **linked** to a single **continuous** **bus** .

(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number : **0 494 056 A2**

(12) # EUROPEAN PATENT APPLICATION

(21) Application number : **91850238.6**

(51) Int. Cl.[5] : **G06F 13/40**

(22) Date of filing : **27.09.91**

(30) Priority : **31.12.90 US 636173**

(43) Date of publication of application :
08.07.92 Bulletin 92/28

(84) Designated Contracting States :
DE FR GB

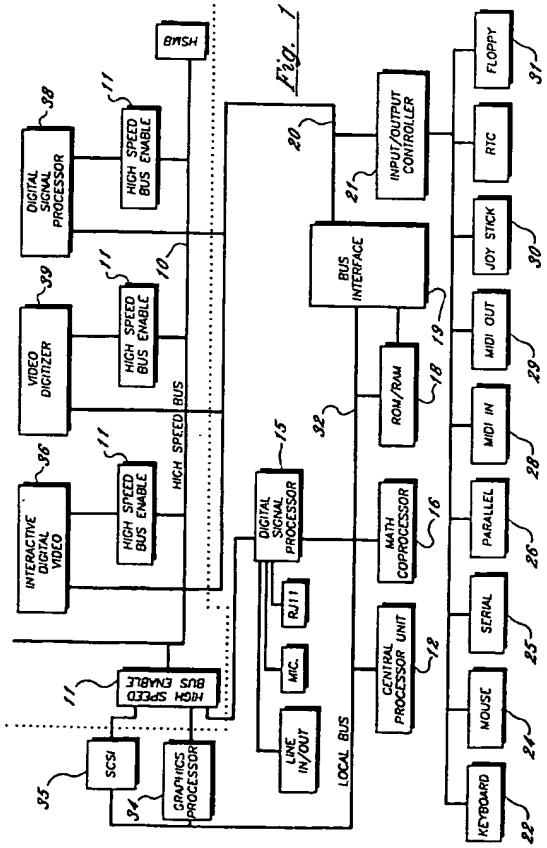
(71) Applicant : **International Business Machines Corporation**
Old Orchard Road
Armonk, N.Y. 10504 (US)

(72) Inventor : **Huynh, Duy Quoc**
2600 Greenwood Terrace, G209
Boca Raton, Florida 33431 (US)
Inventor : **Leverett, Wade Edward**
21471 Saw Mill Court
Boca Raton, Florida 33498 (US)
Inventor : **Tran, Loc Tien**
19107 Farlawn Way
Boca Raton, Florida 33434 (US)

(74) Representative : **Johansson, Lars E. et al**
IBM Svenska AB, Intellectual Property Dept.
S-164 92 Stockholm (SE)

(54) **Dynamically partitionable and allocable bus structure.**

(57) This invention relates to a dynamically partitionable and allocable bus structure for a computer, which facilitates use of the computer in making multi-media presentations. In operation, the computer transfers data among a plurality of data handling devices (12, 15, 34, 35, 36, 38, 39), communicating one with another over a bus (10) which has a plurality of pathway bundles (10A, 10B, 10C), each of which has a plurality of pathways, which has identified steps. The steps include associating with each of the data handling devices a switching device (11) interposed between a corresponding one data handling device and the bus and capable of cooperating with other switching devices for allocating at least one of the bundles to the transfer of data between the respective data handling devices; dynamically selecting and capturing with the cooperating switching devices and from among available bundles that number of bundles needed for transferring data between at least two cooperating switching devices; transferring data through the captured bundles between data handling devices associated with the at least two cooperating switching devices; and releasing captured bundles following completion of data transfer.



EP 0 494 056 A2

39/3,K/117 (Item 117 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00497788
Dynamically partitionable **and allocable** bus **structure.**
Dynamisch unterteilbare und zuteilbare Busstruktur.
Structure de bus fractionnable et attribuable dynamiquement.
PATENT ASSIGNEE:
International Business Machines Corporation, (200120), Old Orchard Road, Armonk, N.Y. 10504, (US), (applicant designated states: DE;FR;GB)
INVENTOR:
Huynh, Duy Quoc, 2600 Greenwood Terrace, G209, Boca Raton, Florida 33431, (US)
Leverett, Wade Edward, 21471 Saw Mill Court, Boca Raton, Florida 33498, (US)
Tran, Loc Tien, 19107 Farlawn Way, Boca Raton, Florida 33434, (US)
LEGAL REPRESENTATIVE:
Burt, Roger James, Dr. et al (52152), IBM United Kingdom Limited Intellectual Property Department Hursley Park, Winchester Hampshire SO21 2JN, (GB)
PATENT (CC, No, Kind, Date): EP 494056 A2 920708 (Basic)
EP 494056 A3 940810
APPLICATION (CC, No, Date): EP 91850238 910927;
PRIORITY (CC, No, Date): US 636173 901231
DESIGNATED STATES: DE; FR; GB
INTERNATIONAL PATENT CLASS: **G06F-013/40**
ABSTRACT WORD COUNT: 191

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS A	(English)	EPABF1	890
SPEC A	(English)	EPABF1	3403
Total word count - document A			4293
Total word count - document B			0
Total word count - documents A + B			4293

Dynamically partitionable **and allocable** bus **structure.**
INTERNATIONAL PATENT CLASS: **G06F-013/40**

...ABSTRACT A2
This invention relates to a **dynamically partitionable** and allocable **bus** structure for a computer, which facilitates use of the computer in making multi-media presentations. In operation, the computer transfers data among a **plurality** of data handling **devices** (12, 15, 34, 35, 36, 38, 39), communicating one with another over a bus (10) which has a **plurality** of pathway bundles (10A, 10B, 10C), each of which has a **plurality** of pathways, which has identified steps. The steps include associating with each of the data handling **devices** a switching **device** (11) interposed between a **corresponding** one data handling **device** and the **bus** and capable of cooperating with other switching **devices** for **allocating** at least one of the bundles to the transfer of data between the respective data handling **devices** ; dynamically selecting and capturing with the cooperating switching **devices** and from among available bundles that **number** of bundles needed for transferring data between at least two cooperating switching **devices** ; transferring data through the captured bundles between data handling **devices** associated with the at least two cooperating switching **devices** ; and releasing captured bundles following completion of data transfer. (see image in

original document)

...SPECIFICATION This invention relates to personal computers, and more particularly to a personal computer having a **dynamically partitionable** and allocable **bus** which facilitates use of the computer in making multi-media presentations.

Personal computer systems in...

...distinguishing characteristics of these systems is the use of a motherboard or system planar to **connect** these **components** together. These systems are designed primarily to give independent computing power to a single user...

...models. In essence, the real mode feature of the 80286, 80386, and 80486 processors provide **hardware** compatibility with software written for the 8086 and 8088 microprocessors.

One growing field of application...

...images derived from television technology, moving image video, audio derived from MIDI (Music Industry Digital **Interface**) files, and the like. Data used in such multi-media presentations or displays typically is relatively expansive, requiring many bits and bytes and a high data **bandwidth** to drive the presentation. Further, where the presentation is to be particularly active (as in...

...object of this invention is to provide, in a multi-media capable personal computer, a **bus** structure which is available for varying **bandwidths** of data transfer **depending** upon the transfers appropriate to various **devices** **coupled** to the **bus** and functioning as source and destination **devices** for various transfers. In realizing this object of the present invention, enhanced multi-media capability is provided in that **bus** resources are **allocated** to the data transfers necessary to effect a desired presentation.

In a further aspect of the invention there is povided a computer capable of high data **bandwidth** transfers in which the transfer of data required to support a presentation such as a multimedia display is facilitated by a capability of **dynamically** **segmenting** and **allocating** **bus** resources to various tasks included in the development of the presentation. In realizing this aspect...

...concurrent transfers between pairs of data sources and data receivers. In realizing this object, the **bus** provided can be **dynamically** **partitioned** to provide **multiple** bundles of pathways, with each bundle **assigned** alone or'in conjunction with others to provide a required data **bandwidth** for one or more types of transfers.

Thus, in accordance with the invention there is provided a **dynamically** allocable **bus** structure for handling communications between a **plurality** of data handling **devices** , each of the **devices** being capable of at least one of receiving and transmitting data, comprising: a **bus** for operatively **coupling** the **devices** one to another, said **bus** comprising a **plurality** of pathway bundles each of which comprises a **plurality** of pathways for data communication; and a **plurality** of switching **devices** , each interposed between a corresponding one of the data handling **devices** and said bus, each of said switching **devices** being operable for cooperating with at least one other switching **device** for allocating to the transfer of data between the respective data handling **devices** at least one of said bundles; said switching **devices** dynamically selecting from among available bundles the **number** of bundles needed for transferring data between cooperating switching **devices** , capturing the requisite **number** of bundles for transfer of

data, and releasing captured bundles following completion of data transfer...

...is also in accordance with the invention provided a method of transferring data among a **plurality** of data handling **devices** communicating one with another over a bus which has a **plurality** of pathway bundles, each of which has a **plurality** of pathways, the method comprising the steps of: associating with each of the data handling **devices** a switching **device** interposed between a **corresponding** one data handling **device** and the **bus** and capable of cooperating with other switching **devices** for **allocating** at least one of the bundles to the transfer of data between the respective data handling **devices** ; dynamically selecting and capturing with the cooperating switching **devices** and from among available bundles that **number** of bundles needed for transferring data between at least two cooperating switching **devices** ; transferring data through the captured bundles between data handling **devices** associated with the at least two cooperating switching **devices** ; and releasing captured bundles following completion of data transfer.
Below a detailed description of a...

...schematic representation of a personal computer embodying the present invention;

Figure 2 is a view **similar** to Figure 1 particularly illustrating the **allocation** of **bus** resources to a particular transfer of data among **devices** **coupled** to an allocable bus in accordance with this invention; and

Figure 3 is a view similar to Figure 2 illustrating a second way in which **bus** resources may be **allocated** in accordance with this invention. Detailed Description of Invention
While the present invention will be...

...the present invention, Figure 1 is a schematic representation of a personal computer having the **dynamically** **segmented** and allocable **bus** of the present invention, while Figures 2 and 3 are schematic illustrations which expand on...

...of the present invention.
Briefly put, the personal computer of the present invention uses a **segmentable** , allocable **bus** 10 to establish independent **data** **pathways** between a **plurality** of data handling **devices** as described more fully hereinafter. Each of the **devices** is capable of at least one of receiving and transmitting data, and so may function as at least one of a source and destination **device** . Many of the **devices** used (as will become more clear hereinafter) are capable of both receiving and transmitting data...

...to time serve both of the functions of being a source and being a destination **device** . **Devices** designed to **attach** to the bus 10 must be able to **couple** to the bus at any point and through switching **devices** 11, each of which is interposed between a corresponding one of the data handling **devices** and the bus 10. The bus 10 has a **plurality** of pathway bundles 10A, 10B, 10C (for example and in Figure 2 and 3), each of which comprises a **plurality** of pathways for data communication. The "bundles" may also be known as "channels". By way...

...10 may have eight bundles of pathways each of which has eight pathways, making a **total** of sixty four **data** conducting **pathways** which may be selectively **segmented** and **allocated** in accordance with this invention. Thus the **bus** 10 may be described as being sixty four bits wide.

Each of the switching **devices** 11 is operable for cooperating with at least one other switching **device** for allocating to the transfer of data between the respective data handling **devices** at least one of the bundles, and thus **segmenting** and **allocating** the **bus** 10. As brought out more fully hereinafter, the switches 11 function for dynamically selecting from among available bundles the **number** of bundles needed for transferring data between cooperating switching **devices** , capturing the requ isite **number** of bundles for transfer of data, and releasing captured bundles following completion of data transfer. **Multiple** data transfer is possible simultaneously along differing paths and directions, with each data transfer being independent of other occurring at that time in other **bus segments** . A **bus segment allocated** for data transfer between two **devices** may be one or up to eight bundles wide **depending** upon the data flow requirements of application software and **hardware** being supported. With a sixty four bit wide capability as here described, the maximum bus **bandwidth** would be 160 Mbytes/second assuming operation at 20 MHz.

The present invention contemplates that the data handling **devices** included in a personal computer in accordance with this invention may take a wide variety of types. The types may include at least one of the **devices** being a central processing unit 12 (Figure 1), at least one of the **devices** being a video processor 34 (Figures 2 and 3), and at least one of the **devices** being a digital signal processor 15 (Figure 1) which, in the disclosed embodiment, functions as an audio and communications processor. As will be understood, such data handling **devices** require varying **bandwidths** for exchange of data, as video data (either compressed or uncompressed) typically requires broader **bandwidths** than audio in order to achieve the necessary transfer rates. It will be understood that **total** transfer rates are related to **bandwidth** due to the correlation of the **numbers** of bits which can be moved in parallel to the **number** of pathways available. The knowledgeable reader will understand that the specific types of **devices** and data transfers here mentioned are presented in order to provide a clear understanding of...

...be associated with a math coprocessor 16, ROM and RAM memory elements 18, a bus **interface** controller 19, and a conventional bus 20 through which are **connected** an input/output controller 21 and a range of input/output (I/O) **devices** . The range of I/O **devices** may include a keyboard 22, pointer **device** 24, serial and parallel ports 25 and 26, MIDI in and out ports 28 and...

...unit 12 may also be associated with a CPU local bus 32 to which are **connected** the audio digital signal processor 15, a graphics processor 34, and a small computer systems **interface** (SCSI) processor 35 for accessing direct access storage **devices** (DASD) such as hard disk storage devices, CD ROM and the like.

Returning to the...

...connecting the respective switch and a corresponding device, and further comprises a plurality of bus **coupling** pathways operatively **connecting** the respective switch and the bus 10. The **number** of bus **coupling** pathways is a **multiple** of the **number** of device **coupling** pathways. The device **coupling** and bus **coupling** pathways are organized into a **plurality** of bundles, with the **bus coupling** bundles **corresponding** in **number** to the **number** of bundles provided in the **bus** 10. AS a consequence, and as described more fully hereinafter, a pair of cooperating switches...

...direction. That is, a group of signals can be programmed to go from a source **device** to a destination **device** , and another group can be programmed to go in the opposite direction.

The switches 11 preferably will be very large scale integrated (VLSI) circuit **devices** functioning as crossbar or **multiplexing** **devices** . As a minimum, the switches 11 must have certain defined characteristics. These characteristics include providing a delay across the **device** which is the same for all lines or pathways in order to assure uniform signal ...

...implementation of the transparent mode described above, each signal pathway must be individually'programmable from **device** to bus **connect** side and vice versa. Each **device** side bundle must be **connectable** with any bus side bundle. When programmed as **connected** with a source **device** in native mode, the switch must output DATA AVAILABLE* and receive DATA ACCEPTED* input. The reverse must be the case for a destination **device** . As will be understood, these two control/status signals are to be bi-directional with...

...which has been received.

The operation of the switches 11 is regulated by a software **interface** here called a **bus** manager. The **bus** manager provides for **allocation** /deallocation of **bus** resources, responds to queries about availability of **bus** resources, and handles error and recovery functions. The design of the bus manager software provides an **interface** for the operating system and application which allows for scaling and expansion of functions as...

...transmission speed; and about bundle status. A query of the first listed type will return **numbers** indicating the **total** **number** of bundles in the system and the **number** presently unused and available. A query of the second listed type will return a **number** defining in whole megahertz values the speed to which the corresponding bundle may be driven...

...scaling bus speed and serves as a mechanism for communicating to an intelligent data handling **device** that data transmission speed may be increased. A query of the third type will return a **number** indicative of whether the corresponding bundle is normal and clocking or in error, with error indications being identified as being alerted by the transmitting or receiving **devices** .

Command functions are contemplated as being allocate bundle; deallocate bundle; and reset. The first identifies...

...operates so as to function in accordance with a'method of transferring data among a **plurality** of data handling **devices** communicating one with another over a bus which has a **plurality** of pathway bundles, each of which has a **plurality** of pathways, which has identified steps. The steps include associating with each of the data handling **devices** a switching **device** interposed between a **corresponding** one data handling **device** and the **bus** and capable of cooperating with other switching **devices** for **allocating** at least one of the bundles to the transfer of data between the respective data handling **devices** ; dynamically selecting and capturing with the cooperating switching **devices** and from among available bundles that **number** of bundles needed for transferring data between at least two cooperating switching **devices** ; transferring data through the captured bundles between data handling **devices** associated with the at least two cooperating switching **devices** ; and releasing captured bundles following completion of data transfer. As will be clear from the discussion above, the step of associating a switching **device** with each data handling device comprises providing a plurality of

bundles of device coupling pathways...

...device coupling bundles and being the same as the number of bundles comprised in said **bus** . Further, the step of **dynamically** selecting and capturing from among available bundles that **number** of bundles needed for transferring data between at least two cooperating switching **devices** comprises selectively interconnecting the bundles of **device coupling** pathways provided by the switch with a **corresponding number** of **bus coupling** bundles provided by the switch.

The possibility of sharing systems resources by **dynamically partitioning** and **allocating** the **bus** 10 interconnecting a **plurality** of data handling **devices** opens the opportunity to reduce redundancy of such resources within a multi-media capable system...

...illustrating the selection and capture of bundles as described above.

More particularly, Figure 2 illustrates **bus segmentation** and concurrent data flow for a full motion video presentation accompanied by sound. Audio data from a SCSI **device** such as a CD-ROM is transferred to an audio digital signal processor 15 for playback. Compressed video data is transferred from the SCSI **device** to a video subsystem 36 for decompression, and the decompressed video is then transferred to...

...video memory requirements otherwise necessary for buffering of the signals are significantly reduced due to **dynamic** sharing of resources.

In Figure 3, **bus segmentation** and concurrent data flow for the digitizing of still video is illustrated. Analog video from...

...vertical synchronization signals, and a blanking signal. Compressed video data is stored in a SCSI **device** using the SCSI controller 35. Two bundles are used for transferring data to the display...

...used for transferring compressed video to the SCSI subsystem.

Each of the examples given in **connection** with Figure 2 and 3 illustrates the flexibility of **bandwidth distribution** among the **devices** accessing the **partitionable** , allocable **bus** 10.

In the drawings and specifications there has been set forth a preferred embodiment of...

...CLAIMS A2

1. A **dynamically** allocable **bus** structure for handling communications between a **plurality** of data handling **devices** (12, 15, 34, 35, 36, 38, 39), each of the **devices** being capable of at least one of receiving and transmitting data, comprising:

a **bus** (10) for operatively **coupling** the **devices** one to another, said **bus** comprising a **plurality** of pathway bundles (10A, 10B, 10C), each of which comprises a **plurality** of pathways for data communication; and

a **plurality** of switching **devices** (11), each interposed between a corresponding one of the data handling **devices** (12, 15, 34, 35, 36, 38, 39), and said bus (10), each of said switching **devices** being operable for cooperating with at least one other switching **device** for allocating to the transfer of data between the respective data handling **devices** (12, 15, 34, 35, 36, 38, 39), at least one of said bundles (10A, 10B, 10C);

said switching **devices** (11) dynamically selecting from among available bundles the **number** of bundles needed for transferring data between cooperating switching **devices** , capturing the requisite **number** of bundles for transfer of data, and releasing captured bundles following completion of data transfer...

...as claimed in claim 1 or 2, wherein said bus comprises eight bundles.

4. A **bus** as claimed in any preceding claim, wherein each of said switching devices comprises a plurality...

...operatively connecting said switch and said bus, and further wherein the number of said bus **coupling** pathways is a **multiple** of the **number** of said device **coupling** pathways.

5. A bus as claimed in any preceding claim, wherein each of said switching devices comprises a **plurality** of bundles of device **coupling** pathways operatively **connecting** said switch and a corresponding device, and further comprises a **plurality** of bundles of bus **coupling** pathways operatively **connecting** said switch and said bus, and further wherein the **number** of said bus **coupling** bundles is a **multiple** of the **number** of said device **coupling** bundles.
6. A bus as claimed in any preceding claim, wherein said switching devices dynamically...

...of sets of paired data handling devices, one of a paired set of data handling **devices** transmitting data and the other receiving transmitted data.

7. A computer incorporating a bus structure...

...any of the preceding claims.

8. A computer as claimed in claim 7, comprising a **plurality** of data handling **devices** and wherein at least one of said **devices** is a central processing unit and at least two of said **devices** are digital signal processors, said data handling **devices** requiring varying **bandwidths** for exchange of data.
9. A computer as claimed in claim 8, wherein at least...

...digital signal processors is an audio processor.

10. A method of transferring data among a **plurality** of data handling **devices** communicating one with another over a bus which has a **plurality** of pathway bundles, each of which has a **plurality** of pathways, the method comprising the steps of:

associating with each of the data handling **devices** a switching **device** interposed between a **corresponding** one data handling **device** and the **bus** and capable of cooperating with other switching **devices** for **allocating** at least one of the bundles to the transfer of data between the respective data handling **devices** ;

dynamically selecting and capturing with the cooperating switching **devices** and from among available bundles that **number** of bundles needed for transferring data between at least two cooperating switching **devices** ;

transferring data through the captured bundles between data handling **devices** associated with the at least two cooperating switching **devices** ; and

releasing captured bundles following completion of data transfer.

11. A method as claimed in claim 10 wherein said step of associating a switching **device** with each data handling **device** comprises providing a plurality of bundles of device coupling pathways operatively connecting the switch and...

...device coupling bundles and being the same as the number of bundles comprised in said **bus** .

12. A method as claimed in claim 11 wherein said step of **dynamically** selecting and capturing from among available bundles that **number** of bundles needed for transferring data between at least two cooperating switching **devices** comprises selectively interconnecting the bundles

of **device coupling** pathways provided by the switch with a **corresponding number** of **bus coupling** bundles provided by the switch.

13. A method as claimed in claim 11 or 12, further comprising **dynamically partitioning** the **bus** into a **plurality** of sets of bundles and concurrently transferring data among a plurality of sets of paired data handling **devices** , one of a paired set of data handling **devices**0 transmitting data and the other receiving transmitted data.
14. A method as claimed in any...

...a coordinated presentation, suitable for achieving coordinated multimedia presentations in a computer system comprising a **plurality** of data handling **devices** including a central processor unit, a video digital signal processor and an audio digital signal processor communicating one with another over a bus which has a **plurality** of pathway bundles, each of which has a **plurality** of pathways. ...

United States Patent [19]

Pope et al.

[11] **Patent Number: 4,797,877**

[45] **Date of Patent: Jan. 10, 1989**

[54] **COMMUNICATION SYSTEM DYNAMIC CONFERENCER CIRCUIT**

[75] Inventors: **Kevin T. Pope, Poway, Calif.; David C. Trimble, Holmdel, N.J.**

[73] Assignees: **American Telephone and Telegraph Company, New York, N.Y.; AT&T Information Systems Inc., Morristown, N.J.**

[21] Appl. No.: **943,242**

[22] Filed: **Dec. 18, 1986**

[51] **Int. Cl.**[4] **H04Q 11/04**

[52] **U.S. Cl.** **370/62; 379/204**

[58] **Field of Search** 370/62; 379/202, 204, 379/205, 206

[56] **References Cited**

U.S. PATENT DOCUMENTS

4,267,593	5/1981	Regan et al.	370/62
4,274,155	6/1981	Funderburk et al.	370/62
4,276,639	6/1981	Fukuda et al.	370/62
4,301,531	11/1981	Lubin	370/62
4,382,295	5/1983	Moffitt et al.	370/62
4,389,720	6/1983	Baxter et al.	370/62
4,466,093	8/1984	Renner	370/62
4,472,798	9/1984	Hafer	370/62
4,479,211	10/1984	Bass et al.	370/62
4,606,021	8/1986	Wurst	370/62

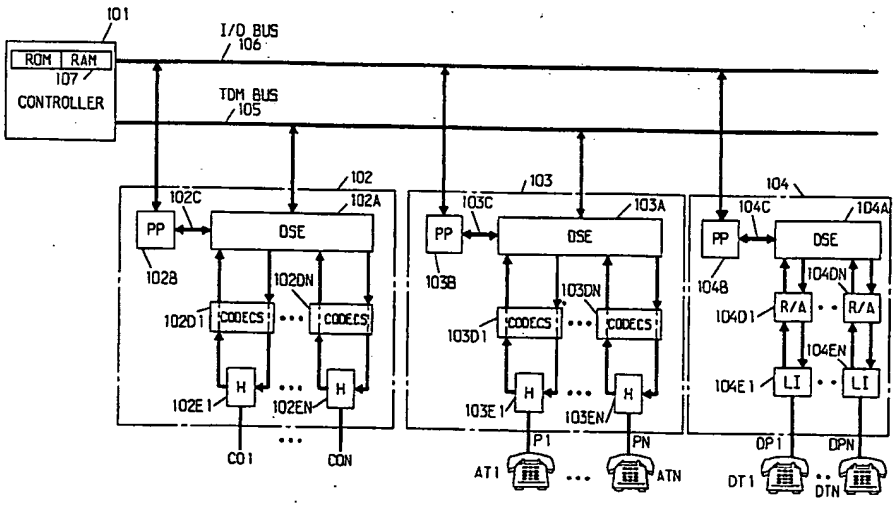
Primary Examiner—Douglas W. Olms

Attorney, Agent, or Firm—John A. Caccuro

[57] **ABSTRACT**

A communication system includes station ports for connecting station sets to the system. A port circuit includes a number of conference registers which are not dedicated to particular station ports but rather are assigned as needed to port circuits. When a request is made to form an N party conference call, N−2 available ones of the conference registers are then assigned to each port of the port circuit which is involved in the resulting conference call connection.

16 Claims, 4 Drawing Sheets



39/3,K/148 (Item 148 from file: 348)
DIALOG(R)File 348:EUROPEAN PATENTS

00273205
Communication system dynamic conferencer circuit.
Dynamische Konferenzschaltung fur Nachrichtensystem.
Circuit de conference dynamique pour systeme de communications.
PATENT ASSIGNEE:
AMERICAN TELEPHONE AND TELEGRAPH COMPANY, (589370), 550 Madison Avenue, New York, NY 10022, (US), (applicant designated states: FR;GB;IT)
INVENTOR:
Pope, Kevin Thomas, 12508 Pintail Court, Poway California 92064, (US)
Trimble, David Carlaw, 17 Winding Brook Way, Holmdel New Jersey 07733, (US)
LEGAL REPRESENTATIVE:
Watts, Christopher Malcolm Kelway, Dr. et al (37392), AT&T (UK) LTD. AT&T Intellectual Property Division 5 Mornington Road, Woodford Green Essex IG8 OTU, (GB)
PATENT (CC, No, Kind, Date): EP 272828 A2 880629 (Basic)
EP 272828 A3 890705
EP 272828 B1 930303
APPLICATION (CC, No, Date): EP 87310723 871207;
PRIORITY (CC, No, Date): US 943242 861218
DESIGNATED STATES: FR; GB; IT
INTERNATIONAL PATENT CLASS: H04M-003/56; H04Q-011/04;
ABSTRACT WORD COUNT: 52

LANGUAGE (Publication,Procedural,Application): English; English; English
FULLTEXT AVAILABILITY:

Available Text	Language	Update	Word Count
CLAIMS B	(English)	EPBBF1	850
CLAIMS B	(German)	EPBBF1	710
CLAIMS B	(French)	EPBBF1	1053
SPEC B	(English)	EPBBF1	5464
Total word count - document A			0
Total word count - document B			8077
Total word count - documents A + B			8077

...ABSTRACT which are not dedicated to particular station ports (P1-PN) but rather are pooled and **assigned as needed** to any port requesting to conference a party to an existing **connection** . (FIG.2) ...

...SPECIFICATION relates to a digital conferencing arrangement having a common pool of circuitry which can be **dynamically allocated** to one or more conference **connections** .
Background of the Invention
Distributed digital **conference** systems enable individual system ports to operate under processor control to create a conference **connection** by combining selected time slot digital **port** signals. In such systems, to assure that the **number** of conference registers required for a **conference connection** was sufficient, a large fixed **number of** conference registers were dedicated to **each** station port Consequently, when a conference **connection** involved less than the maximum **number** of conferees, there were many idle conference registers. With the continuing increase in complexity of...

...ports. Thus, for example, according to the present invention a port circuit board which interfaces **X** ports (where X is a **number**) may include a pooled group of X registers which can **be** dynamically

assigned to **establish** an X+2 party **conference connection** , or individually assigned in some distribution to more than one conference **connection** . Since the **number** of registers provided on each port **circuit** board would typically be much **less** than the maximum **number** of **connections** expected **per** conference multiplied by the **number** of ports serviced by the port board, there is the potential for conference blocking. Consequently...

...number of ports in a conference connection in order to maintain acceptable blocking characteristics.

Brief **Description** of the Drawing

In the drawings,

FIG. 1 is a block diagram of a communication...

...illustrate the contents of various registers and buffers utilized by the present invention; and

FIG. **4** illustrates a representative timing **diagram** useful in describing **the** operation of the present invention.

System Description

While **the** disclosed embodiment is described for use in a telephone system, it more generally can be used in any digital arrangement for combining a **plurality** of digital inputs in **dynamically** defined combinations and **distributing** the resulting digital signal to any of a **plurality** of **outputs** . Such application may include use in a digital paging system, a radio system, a multi...

...following description, each element of each figure has a reference designation associated therewith, the first **number** of which refers to the figure in which that element is located (e.g., 101 is located in FIG. 1).

Shown in FIG. **1** is an illustrative block diagram of a digital communication system useful in describing the operation...

...control 101 which interconnects to port boards 102, 103, and 104 via a time-division **multiplexed** (TDM) bus 105 and an input/output (I/O) bus 106. Controller 101 establishes and controls communications over TDM bus 105 and I/O **bus** 106. Program memory 107 **may include** random- **access** -memory (**RAM**) **and** read-only-memory (**ROM**) to provide stored program instructions to controller 101 for controlling the operation of various communication features and functions of the system. Controller 101 **includes** a standard arrangement of a microprocessor; **a real time** clock; **periodic interrupt circuitry** ; **port** board and other **interface** circuitry; clock and tone circuitry; and associated power-up/power-down, reset and sanity timer...

...I/O bus 106 includes a 16-bit address bus, an 8-bit data bus, **multiple** port board control signals, a reset signal, a clock signal, a frame reset signal, and...

...signal. The I/O bus communicates control signals to control the information flow between controller **101** and port boards 102, 103 and 104.

The TDM bus 105, for example, may operate 106 directly and also connects to TDM bus 105 directly. Port processors 102B - 104B **connect** via I/O bus 106 to controller 101. Controller 101 controls the function of each **port board** 102 - 104 via the port processors 102B - 104B. In an alternative embodiment, the function performed by port processors 102B - 104B **can be** incorporated into controller 101.

Illustratively, port board 102 utilizes one DSE 102A and one port processor 102B to **interface multiple** analog central office (CO) lines CO1 - CON via separate hybrids **102E1 - 102EN** and codecs 102D1 - 102DN. Similarly, port board 103 **interfaces multiple analog**

telephones AT1 - ATN, via ports P1 - PN, separate hybrids 102E1 - 103EN and codecs 103D1 -103DN. Illustratively, port board 104 utilizes one DSE 104A and one port processor 104B to **interface** **multiple** digital telephones DT1 - DTN via ports DP1 - DPN, separate line **interfaces** (LI) 104E1 - 104EN and rate and adaption (R/A) circuits 104D1- 104DN. For purposes of the present description, it is assumed that N is **equal** to 16.

Before proceeding with the detailed operating description of the present invention, it should...

...may be utilized in other telephone communication systems. Since such systems utilize a variety of **hardware** and programs used to control the communication system. However, the present invention must be incorporated ...

...or utilize other encoding algorithms and so is not a requirement to implement the present **invention** . Each DSE 102A, 103A and 104A **contains** **a** **portion** **of** the switch matrix of the system. These DSEs communicate digital signal (e.g., 8-bit voice samples) and control data between **each** other and controller 101 via **the** 8-bit TDM **bus** 105. **As** previously noted, the **DSEs** **interface** **multiple** port **devices** --e.g., codecs (such as 102D1) or digital transceivers (e. **g** ., digital microphones and receivers such as DT1)--to the TDM **bus** 105. The DSE places port device samples onto the removes **port** device samples from TDM **bus** 105 during specific time slots under control of controller 101. Thus, two time slots **per** call are needed, **one** time **slot** for transmitting **and** one for receiving.

With continued reference to FIG. 2, **port** **interface** 201 **interfaces** to DSE 103A a **total** of N devices 103D1 to 103DN which may include **codecs** (**e** .g., **103D**) and/or digital transceivers (**e** .g., **digital** **telephone** DT1). Port **interface** 201 utilizes a standard transmission format for digital serialized voice and control data transmission and reception from the codecs and/or digital transceivers.

On the system side, bus **interface** 202 **interfaces** DSE 103A to TDM bus 105. Bus **interface** 202 includes transmit sample buffer 203, receive sample buffer 205 and time slot match detector...

...processor bus 103C and stored in time slot memory 236, accesses specific time slots out **of** the 256 time slots on **TDM** bus 105. With reference to TS Table 300 of FIG. 3, time slots are **associated** **with** each of the transmit (TX) **channels** 301 and receive (RX) channels 302 for the N ports (illustratively, N = 16) **connected** to DSE 103A via port **interface** 201 and are associated with each of the conference registers 303 of DSE 103A **as** **needed** . The commands received over processor **bus** 103C originate from controller 101. Returning to FIG. 2, transmit buffer 203 stores the 16....

...port is identified with a conference register. The conference signal samples together with the appropriate **received** port signals (RX-TSA) are sent to accumulator 206 to generate a combined received conference...

...receive buffer 205 to "(mu)" or "A"-law samples for output to particular codecs via **port** **interface** **201** .

Conference slot address sequencer **207** , together with TS detector 204, receive buffer 205, accumulator 206, and conference slot allocation memory 212, enables DSE 103A to provide the disclosed dynamic conferencing **capability** .

In accordance with the present invention, DSE 103A **allocates** or **assigns** conference registers to port **dynamically** under control of **controller** 101. The conference register to port assignments are stored

in conference memory 212. Illustratively, as previously assumed, DSE 103A has a **total** of 16 conference registers and **services** 16 station port (i.e., N = 16). Thus, the conference registers can be assigned to...

...i.e., P1 through P16) in any combination. To conference a party to an existing **connection** , a conference register or slot is assigned to store that party's received signal ...Thus, generally, for an "M" party conference, "M - 2" conference registers or slots must be **assigned** to each participating station **port** of DSE 103A. In this fashion, multi-party conferences are supported via a shared conference...

...conference call (M = 5) will be described in a later paragraph.

DSE 103A is programmed **by** controller 101 via processor **bus** 103C. Time slot memory 236 is programmed with the transmit and receive time slot assignments...

...conference registers C1, C1 and C3 are assigned to port P1 and conference registers C4, **C5** and C6 are **assigned** to port P2 as shown by **Table** 310 of FIG. 3. Note, the **particular assignment** of conference registers and time slots are for purposes of illustration only.

With joint reference to FIGS. 2 and 3, conference slot **allocation** memory 212 is also programmed by controller 101 via processor **bus** 103C. The contents of conference slot **allocation** memory 212 is shown in Table 310 which shows the assignment of conference registers to ports. In the disclosed embodiment, Table 310 includes 16 **registers** or words, each 4 bits long. Each conference register corresponds to one of the 16 available conference slots. These conference slots are individually **allocatable** to ports in any combination. A conference slot is **allocated** by **a port by** writing the **port** number (P0 **to** P15) **into** the desired conference register. This 16-word, 4-bit per word conference memory 212 may be implemented as a form of associative memory. **This** conference memory is used to **configure** the **dynamic** conferencer circuit of FIG. 2 **for** the **assignment** or allocation of conference registers.

Illustratively, for the 5-party conference call examples, as shown...

...are assigned to ports P1 and P2, respectively, indicating that ports P1 and P2 are **connected** to a 5-party conference call. With reference to 303 of Table 300, registers C1...

...In this illustrative 5-party conference call, the party at port P1 would receive the **8** -bit voice sample TS **16** arithmetically summed by accumulator 206 with the voice samples associated with the time slots (TS ...

...specified by registers C1 - C3. This summed 4-party voice signal is outputted via port **interface** 201 to the facility **connected** to port P1. Similarly, the party at port P2 receives the summed sample from TS 3, TS 4, TS 5 and TS 6. The remaining conferees would, **in** this particular embodiment, receive their summed samples via either port circuit board 102 if the...

...104 if the party is on a digital station set. Whether the remaining conferees were **connected** via port board 102 or port board 104, the operation would be analogous to that...

...step is to load the 32-word, 8-bits per word receive buffer 205 with **samples** from TDM bus 105. Table 320 illustrates the content of receive buffer 205. As previously...

...addresses into the 48-word,

8-bit time slot memory 236. Whenever the time slot **number** of time slot counter 213 matches the contents of one or more of the port...

...signal 255 causes a sample from transmit buffer 203 to be loaded onto TDM bus **105** . In this manner, a new set of TDM bus 105 samples gets placed into receive...

...frame a new set of transmit samples is loaded into transmit buffer 203 from port **interface** 201 via bus 225.

The 32-word, 8-bit receive buffer 205 may be implemented...or more of the row select inputs can be active at any time.

Similarly, the **16** -word, 8-bit transmit buffer 203, contents illustrated by Table 330, may be implemented as...

...only and the other an output-only port. Samples received over bus 225 from port **interface** 201 are loaded into transmit buffer 203 via the input ports. The 16 row select...

...slot is to be taken from or inserted on TDM bus 105.

Table 320 illustrates **the** contents of receive buffer 205 for **the** 5 -party conference call example **being** discussed. The samples contained in the bottom 16 locations of receive buffer 205 are sent out in sequence every frame to each of the 16 ports via port **interface** 201. The top 16 locations are **dynamically allocated** to ports by the **dynamic** conference **circuit** using Table **310** . For each port, the associated conference samples are extracted from the top 16 locations of...

...the bottom part of receive buffer 205.

With reference to FIGS. 2, 3 and 4, **the** operation of conference slot address sequencer 207 is described. Conference slot address sequencer 207 has...

...buffer 205 which are to be combined to form the conference signal for each port **connected** to DSE 103A. Briefly, with reference to TS Table 300 for the 5-party conference...

...port P2. Additionally, the user at port P1 receives the other three conferee voice signal **samples** TS 3, TS 4 and TS 5 from, respectively, conference registers C1, C2 **and** C3. Accumulator 206 combines the voice sample signals (TS 16, TS 3, TS 4 and TS 5) and **outputs** the resultant combined signal to port P1 via **interface** 201. Of course, the user at port P1 also receives a side tone **component** of his/her own voice at the station set. Similarly, the user at port P2, who is also **connected** to DSE 103A, receives the voice signal sample TS 6 from port P1 in **addition** to voice signal samples TS 3, TS 4 and TS 5 from, respectively, conference registers...

...DSE 103A services each of its 16 ports in sequence once every frame. The port **number** of the port currently being serviced is contained in port counter 222. Port counter 222...

...conference memory 212. As shown by Conference Assign Table 310, conference slots C1 - C3 are **assigned** to port P1. Conference memory **212** is a content addressable memory. Thus, when port address P1 is inputted into conference memory...shown in FIG. 4.

Since the first three conference registers C1, C2 and C3 are **assigned** to **port P1** , their output leads set the set/reset (S/R) flip-flop 228 of the first...

...the voice sample from conference register C3 is sent to accumulator 206,

the voice sample **from** conference **register** C2 and then conference register C1 will be sent to accumulator 206.

The 4-bit...

...and the 4-bit output 231 of port counter 222 are inputs to two-channel **multiplexer** 223. **Multiplexer** 223 selects input 230 when Done signal (lead 231) is at logic 0 and selects...

...buffer address bus 224 that is used to extract voice samples from receive buffer 205 **in** the proper order. The samples are fed to accumulator circuit 206 which combines the samples...

...the inverted Done signal 231 as the most significant bit and the output 225 of **multiplexer** 223 as the lower four bits.

With reference to **FIG** . 4, the Done signal 231 is generated by encoder 221 when no more of its...

...16) are active or logic 1. This indicates that no conference register remains to be **serviced** for the current port **number** . Thus, Done signal 231 is at logic 0 while the voice samples of conference registers...

...processed. The Done signal 231 is used to enable port counter 222 and to switch **multiplexer** 223.

During time t1, the address 0010 from encoder 221--representing conference register C3--together...

...logic 1 of the inverted Done lead 231 to form the address 10001 on address **bus** 224. Note Done lead **231** remains at logic 0 since conference register S/R flip-flop 217-1 is still...

...and TS 4, respectively, from registers C3 and C2.

At time t4, the next port **clock** signal 233 causes decoder 220 to clear S/R flip-flop 217-1. Since there...

...a Done signal 231 at logic 1. The Done signal 231 at logic 1 causes **multiplexer** 223 to switch to accept the port count output 235 from port counter 222. The **by** Table 320, the address 100000 accesses the receive voice sample TS 16 for receive port...

...received samples TS 3, TS 4 and TS 5. This sum is outputted via port **interface** 201 to the user at port P1.

At time t5, the next port clock signal...

...226 to conference memory 212. As shown by Table 310, port P2 has conference registers **C4** , C5 and C6 assigned thereto. The outputs of conference memory 212 associated with C6, C5...

...flops 217-6, 217-5 and 217-4 (not shown) to set. Again, the highest **number** of **S** /R flip-flop 217-6 (conference register C6) is binarily encoded (0101) by encoder 221...

...encoder 221 are at logic 1, Done signal lead 231 goes to logic 0 causing **multiplexer** 223 to accept inputs on bus 230 from encoder 221.

In a sequence essentially the...

...TS 4, TS 3 and TS 6 associated, respectively, with conference registers C6, C5 and **C4** and receive port P2 register (P2R of Table 320) from receive sample buffer 205. This...

...t9. Accumulator 206 combines the samples into one voice signal which is

outputted via port **interface** 210 to the user at port P2.
At time t9, the next port clock signal...

...Enable signal to become logic 1, thereby enabling conference slot address sequencer 207 to generate **the** samples required by ports P1 to P16 during the new frame.
As previously noted, accumulator...

...retrieved from the receive sample buffer 205 and to send the sum to the port **interface** 201 for output to the appropriate station port. Since the samples for a given port...
...or "A"-law encoded form. A front-end expander circuit 206-1 converts the samples **into** a 14-bit linear form. The linear values are summed with the contents of the...

...In the formation of a conference sum, the first port sample is loaded directly via **multiplexer** 206-3 into the accumulator register 206-4. Subsequent samples associated with that port are...

...the accumulator register 206-4. Thus, after all samples associated with a given port have **been** retrieved, the accumulator register 206-4 contains the sum of all **previous** samples. The 16-bit linear sum is converted by compressor 206-5 to 8-bit "(mu)" or "A" compressed form and shifted out of the DSE port via port **interface** 201.
While the present embodiment describes a request for a conference **connection** as having originated from system ports, it likewise could be initiated by the system attendant...

...the teachings of the present invention. Additionally, while the application of the present invention has **been** described for use with digitized **voice** signals, its operation using digitized video, graphic signal or other data signal is contemplated as...
...present invention. Moreover, while the embodiment of the present invention is described as using particular **hardware** (i.e., controllers, etc.), bus **sizes** and data rates, it should be recognized that equivalents could be utilized without affecting the...

...CLAIMS B1

1. A **port circuit** including a **plurality** of ports (102, 103) **each** port **interfacing a** digital **device** to a digital communication **system , said port circuit** comprising
   a **plurality** of conference registers (in 205) for storing digital port signals, wherein each of said conference registers is assignable to any **of** said **plurality** of ports; CHARACTERISED IN THAT the **plurality** of conference **registers** is pooled as a common resource and **in** that the **circuit** comprises
   assigning means (**110** ,212) responsive to a signal requesting an N party conference **connection** (N is an integer >2) **for assigning** no more than N-2 **available ones** out of the whole pool of said conference registers to each port of **said** port circuit **connected** to said N party conference **connection** .
2. The port circuit of claim 1 further comprising
   a receive sample buffer (205) including a **plurality** of receive registers for storing digital port signals **received** by said **port circuit** and wherein **said receive** sample buffer includes said conference registers.
3. The **port circuit** of claim 1 wherein said conference registers include a dual-port memory.
4. The port circuit of claim 1 wherein said **assigning means** includes

means for identifying a port **assignment** for each of said conference registers.

5. The **port circuit** of claim 1 further comprising

a time slot counter (**213**) for counting each digital **port** signal time slot at said port circuit;

a receive sample buffer (205) including a **plurality** of **receive** registers, each of said receive **registers** arranged to store a digital port signal received during an existing **connection** of one of said **plurality** of ports; and

a time slot match detector (236,204) including

means for associating (236) time slots with said **plurality** of receive registers and said **plurality** of conference registers, and

means (204) responsive to said time slot counter and said associating means for **enabling** one or more of said receive registers and said conference registers to store a digital port signal associated with the **count** of said time slot counter.

6. The port circuit of claim 5 further comprising

a transmit sample **buffer** (203) including a **plurality** of transmit registers, each of said transmit registers arranged to store a digital port signal transmitted from **one** of said **plurality** of ports;

wherein said associating means of said time slot match detector associates time slots with **said plurality** of transmit registers; and

wherein said enabling means of said **time** slot match detector enables an output from one of said **plurality** of transmit registers of a digital port signal associated with the count of said time slot counter.

7. The **port circuit** of claim **1** further comprising

a receive register (205) **assigned** to said requesting port for storing a digital **port** signal **received thereat** ,

conference slot sequencer means (207) for accessing digital port signals stored in said conference registers...

...receive register assigned to said requesting port, and

accumulating means (206) responsive to said sequencer **means** for summing said accessed digital port **signals** .

8. The **port circuit** of claim 7 wherein said sequencer means includes

means for generating (216) an address for...

...register and

means (221) utilizing said address for sequentially selecting for access each conference register **assigned** to a **port** involved in a **connection** .

9. The **port circuit** of claim 8 wherein said sequencer means further includes

means for generating (221, 231) a done signal when said sequential selecting means has completed selection of conference registers **assigned** to said port involved in said **connection** , **and**

port counting means (222) for determining a port being serviced by said sequencer means, said port...

...of said done signal for switching said conference register address from said asssigning means to **access** a conference register and responsive to the presence of said done signal for switching a port count address from said port **counting** means to access a receive register.

11. The port circuit of claim 9 further comprising

a receive sample buffer (205) including a **plurality** of receive

registers or storing digital port signals received by said **port** circuit and wherein said receive sample buffer includes a **plurality** of conference registers and wherein

said sequencer **means** (207) accesses said receive sample buffer using said done signal as the most significant bit of said address and uses said port count address to address said **plurality** of receive registers when said done signal is at 1, state and to address said **plurality** of conference registers when said done signal is at a second 1, state.

12. The...

...clock signal (233) different from a clock used by said time slot counter.

13. The **port circuit** of claim 1 wherein said **assigning** means (212) is responsive to said requesting signal received from one of said **plurality** of ports.

14. The **port circuit** of claim 13 wherein said **assigning** means (212) is responsive to said requesting signal received from a port of said **plurality** of ports which is not a party to said existing **connection** .

15. The **port circuit** of claim 13 wherein said **assigning** means (212) is responsive to said requesting signal received from a port of said **plurality** of ports which is a party to said existing **connection** .

US006662260B1

(12) **United States Patent**
Wertheim et al.

(10) **Patent No.: US 6,662,260 B1**
(45) **Date of Patent: Dec. 9, 2003**

(54) **ELECTRONIC CIRCUITS WITH DYNAMIC BUS PARTITIONING**

(75) Inventors: **Elon Wertheim**, Hod Hasharon (IL); **Andreas Daniel Olofsson**, Tel Aviv (IL)

(73) Assignee: **Analog Devices, Inc.**, Norwood, MA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/537,066**

(22) Filed: **Mar. 28, 2000**

(51) **Int. Cl.**[7] **G06F 13/00**
(52) **U.S. Cl.** **710/316**
(58) **Field of Search** 710/305–317, 710/300

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,821,174 A		4/1989	Webb et al.	
4,922,409 A		5/1990	Schoellkopf et al.	
5,862,359 A	*	1/1999	Nozuyama	710/316
5,896,543 A		4/1999	Garde	395/800.35
6,081,863 A	*	6/2000	Kelley et al.	710/312
6,182,178 B1	*	1/2001	Kelley et al.	710/107
6,195,723 B1	*	2/2001	Neal et al.	710/113
6,237,057 B1	*	5/2001	Neal et al.	710/309
6,338,107 B1	*	1/2002	Neal et al.	710/302

FOREIGN PATENT DOCUMENTS

EP 0 892 352 1/1999

OTHER PUBLICATIONS

Kapadia et al., "Reducing Switching Activity on Datapath Buses With Control-Signal Gating", IEEE Journal of Solid-State Circuits, vol. 34, No. 3, (Mar. 1999), pp. 405–414.

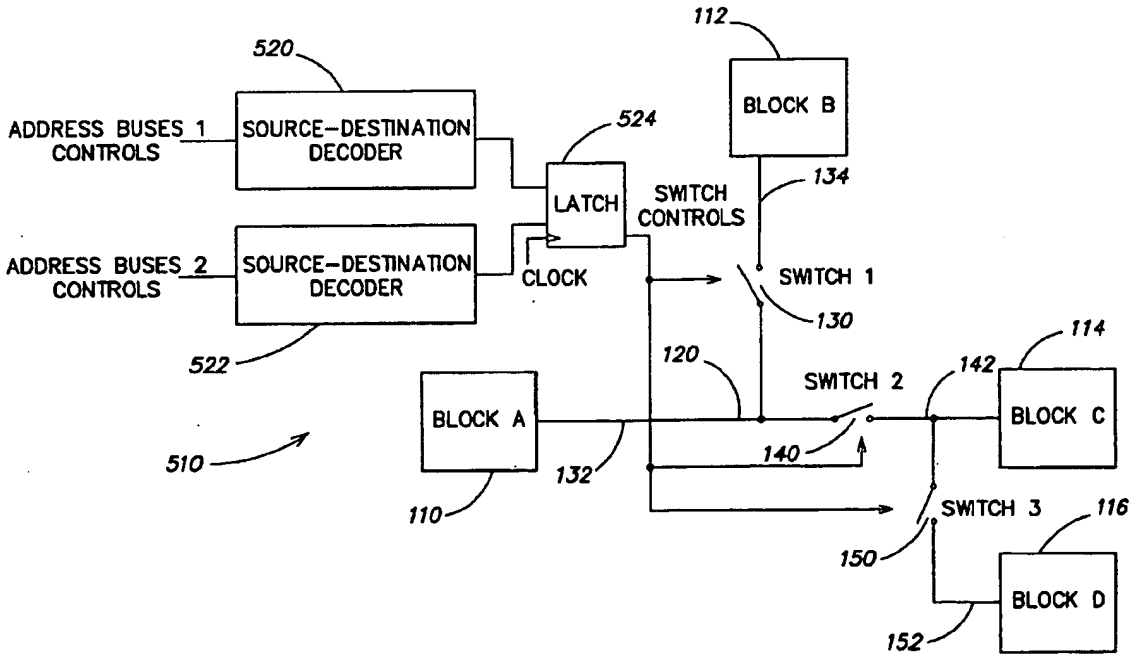
* cited by examiner

Primary Examiner—Paul R. Myers
(74) *Attorney, Agent, or Firm*—Wolf, Greenfield & Sacks, P.C.

(57) **ABSTRACT**

Electronic circuits utilize dynamic bus partitioning to reduce power consumption. An electronic circuit includes several functional electronic blocks, a bus interconnecting the functional blocks, one or more electronically controllable switches partitioning the bus into bus segments and a switch controller. Each of the electronically controllable switches has an on state wherein two of the bus segments are interconnected and an off state wherein the two bus segments are isolated. The switch controller controls the states of the switches in response to control information representative of the source and the destination of each bus transaction. The switch controller may dynamically change the states of the switches between transactions of a sequence of bus transactions. In another embodiment, the switch controller may control the states of the switches to permit two or more bus transactions to be performed simultaneously.

25 Claims, 7 Drawing Sheets



What is claimed:

1. Electronic apparatus comprising:

a plurality of functional electronic blocks on a single monolithic integrated circuit;

an on-chip bus interconnecting said functional blocks;

one or more on-chip electronically controllable switches partitioning said bus into bus segments, each of said electronically controllable switches having an on state wherein two of said bus segments are interconnected and an off state wherein said two bus segments are isolated, each of said electronically controllable switches comprising a dynamic logic bidirectional bus buffer; and

an on-chip switch controller for controlling the states of said electronically controllable switches in response to control information, wherein said functional blocks, said bus, said electronically controllable switches and said switch controller are fabricated on a single substrate.

2. Electronic apparatus as defined in claim 1, wherein said functional blocks are components of a digital signal processor.

3. Electronic apparatus as defined in claim 1, wherein said functional blocks comprise a computation block, a memory block and a control block.

4. Electronic apparatus as defined in claim 1, wherein said bus comprises multiple conductors and wherein each of said electronically controllable switches partitions each conductor of said bus.

5. Electronic apparatus as defined in claim 1, wherein said switch controller comprises a source-destination decoder for controlling the electronically controllable switches in response to control information representative of the source and the destination of each bus transaction.

6. Electronic apparatus as defined in claim 1, wherein said switch controller comprises means for dynamically changing the states of said electronically controllable switches between transactions of a sequence of bus transactions, in response to control information representative of the source and the destination of each of the bus transactions.

7. Electronic apparatus as defined in claim 1, wherein said switch controller comprises means responsive to control information representative of the source and the destination of each bus transaction for controlling the states of said electronically controllable switches.

8. Electronic apparatus as defined in claim 1, wherein said switch controller comprises means for controlling the states of said electronically controllable switches to permit two or more simultaneous bus transactions on said bus.

9. Electronic apparatus comprising:

a plurality of functional electronic blocks;

a bus interconnecting said functional blocks;

one or more electronically controllable switches partitioning said bus into bus segments, each of said electronically controllable switches having an on state wherein two of said bus segments are interconnected and an off state wherein said two bus segments are isolated; and

a switch controller for controlling the states of said electronically controllable switches in response to control information, wherein said switch controller comprises first and second source-destination decoders, each responsive to control information representative of the source and the destination of a different bus transaction, for controlling the states of said electronically controllable switches to permit first and second simultaneous bus transactions on said bus.

10. Electronic apparatus as defined in claim 1, wherein each dynamic logic bidirectional bus buffer comprises, in each bus direction, a dynamic logic NOR gate and a field effect transistor.

11. In electronic apparatus comprising a plurality of functional electronic blocks interconnected by a bus, a method for communicating between the functional blocks, comprising the steps of:

fabricating a plurality of functional electronic blocks on a single monolithic integrated circuit;

interconnecting the functional electronic blocks with an on-chip bus;

partitioning the on-chip bus into bus segments using electronically controllable dynamic logic bidirectional bus buffers;

enabling a bus transaction between a source functional block and a destination functional block by interconnecting bus segments to complete a connection between the source and destination functional blocks; and

performing the bus transaction on the interconnected bus segments.

12. A method as defined in claim 11, wherein the step of partitioning the bus comprises partitioning the bus with one or more electronically controllable switches and wherein the step of enabling a bus transaction comprises controlling the states of the electronically controllable switches in response to control information representative of the source and the destination of the bus transaction.

13. A method as defined in claim 11, wherein the step of partitioning the bus comprises partitioning the bus with one or more electronically controllable switches and wherein the step of enabling a bus transaction comprises dynamically changing the states of said electronically controllable switches between transactions of a sequence of bus transactions, in response to control information representative of the source and the destination of each of the bus transactions.

14. A method as defined in claim 11, wherein the step of partitioning the bus comprises partitioning the bus with one or more electronically controllable switches and wherein the step of enabling a bus transaction comprises controlling the states of said electronically controllable switches to permit two or more simultaneous bus transactions on said bus.

15. A method as defined in claim 11, wherein the step of enabling a bus transaction further comprises the step of electrically isolating bus segments that are not needed for the bus transaction.

16. A method as defined in claim 11, wherein each of the electronically controllable dynamic logic bidirectional bus buffers comprises, in each bus direction, a dynamic logic NOR gate and a field effect transistor.

17. Electronic apparatus comprising:

a plurality of functional electronic blocks on a single monolithic integrated circuit;

an on-chip bus interconnecting said functional electronic blocks, wherein said functional electronic blocks and said bus are fabricated on a single substrate;

means for partitioning the on-chip bus into bus segments, comprising electronically controllable dynamic logic bidirectional bus buffers; and

means for enabling a bus transaction on the bus by interconnecting bus segments in response to control information representative of the source and the destination of the bus transaction.

18. Electronic apparatus as defined in claim 17, wherein said functional blocks are components of a digital signal processor.

39/3,K/190 (Item 190 from file: 349)
DIALOG(R)File 349:PCT FULLTEXT

00839908 **Image available**
ELECTRONIC CIRCUITS WITH DYNAMIC BUS PARTITIONING
CIRCUITS ELECTRONIQUES A PARTITIONNEMENT **DYNAMIQUE DU** BUS
Patent Applicant/Assignee:
ANALOG DEVICES INC, One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, US, US (Residence), US (Nationality)
Inventor(s):
WERTHEIM Elon, Habariach 7, Hod Hasharon, IL,
OLOFSSON Andreas Daniel, Pasternack 3, Apartment 50, 61650 Tel Aviv, IL,
Legal Representative:
MCCLELLAN William R (agent), Wolf, Greenfield & Sacks, P.C., 600 Atlantic Avenue, Boston, MA 02210, US,
Patent and Priority Information (Country, Number, Date):
Patent: WO 200173566 A2-A3 20011004 (WO 0173566)
Application: WO 2001US9521 20010326 (PCT/WO US0109521)
Priority Application: US 2000537066 20000328
Designated States:
(Protection type is "patent" unless otherwise stated - for applications prior to 2004)
JP
(EP) AT BE CH CY DE DK ES FI FR GB GR IE IT LU MC NL PT SE TR
Publication Language: English
Filing Language: English
Fulltext Word Count: 5860

ELECTRONIC CIRCUITS WITH DYNAMIC BUS PARTITIONING
CIRCUITS ELECTRONIQUES A PARTITIONNEMENT **DYNAMIQUE DU** BUS
Main International Patent Class: **G06F-013/40**
Fulltext Availability:
Detailed Description
Claims

English Abstract
Electronic circuits utilize **dynamic bus partitioning** to reduce power consumption. An electronic circuit includes several functional electronic blocks, a **bus** interconnecting the functional blocks, one or more electronically controllable switches **partitioning** the **bus** into **bus segments** and a switch controller. Each of the electronically controllable switches has an on state wherein two of the **bus segments** are interconnected and an off state wherein the two **bus segments** are isolated. The switch controller controls the states of the switches in response to control information representative of the source and the destination of each **bus** transaction. The switch controller may **dynamically** change the states of the switches between transactions of a sequence of bus transactions. In...

French Abstract
La presente invention concerne des circuits electroniques utilisant le **partitionnement** dynamique du **bus** pour reduire la consommation d'electricite. Un tel circuit electronique se compose de plusieurs blocs ...

...d'un bus interconnectant les blocs fonctionnels, d'un ou de plusieurs commutateurs electroniquement commandes **partitionnant** le **bus** en **segments** de **bus** , et un controleur de commutation. Chacun de ces commutateurs electroniquement commandes est soit ferme pour

interconnecter deux des **segments** du **bus** , soit ouvert pour isoler ces deux **segments** du **bus** . Le controleur de commutation commande l'etat des commutateurs en reaction a de l'information...

Detailed Description
ELECTRONIC CIRCUITS WITH **DYNAMIC**
BUS PARTITIONING
Field of the Invention
This invention relates to buses used for carrying information between functional blocks of an electronic circuit and, more particularly, to electronic circuits which utilize **dynamic bus partitioning** for reduced power consumption. The invention is typically utilized for improving the operation of high...

...electronic apparatus is provided. The electronic apparatus comprises a plurality of functional electronic blocks, a **bus** interconnecting the functional blocks, one or more electronically controllable switches **partitioning** the **bus** into **bus segments** and a switch controller. Each of the electronically controllable switches has an on state wherein two of the **bus segments** are interconnected and an off state wherein the two **bus segments** are isolated. The switch controller controls the states of the electronically controllable switches in response...

...switches in response to control information representative of the source and the destination of each **bus** transaction. The switch controller may dynamically change the states of the electronically controllable switches between...

...plurality of functional electronic blocks interconnected by a bus. The method comprises the steps of **partitioning** the bus into bus **segments** , enabling a bus transaction between a source functional block and a destination functional block by interconnecting **bus segments** to complete a **connection** between the source and destination functional blocks, and performing the bus transaction on the interconnected bus **segments** .

According to a finther aspect of the invention, electronic **apparatus** comprises a **plurality** of functional electronic blocks interconnected by a bus, means for **partitioning** the bus into bus **segments** , and means for enabling a **bus** transaction on the **bus** by interconnecting **bus segments** in response to control information representative of the source and the destination of the bus...

...block diagram of a first example of an implementation of an electronically
controllable switch for **partitioning** a **bus** into **bus segments** ; Fig. 8 is a block diagram of a second example of an implementation of an electronically controllable switch for **partitioning** a **bus** into **bus segments** ; and Fig. 9 is a block diagram of an embodiment of the invention wherein two **bus** transactions may be performed on different groups of **bus segments** simultaneously.

Detailed Description
A block diagram of a digital signal processor (DSP) IO suitable for...

...120 is a multiple conductor connection between the electronic circuits.

An electronically controllable switch 130 **partitions bus** 120 into a **bus segment** 132 and a **bus segment** 134. An electronically controllable switch 140 **partitions bus** 120 into **bus segment** 132

and a **bus segment** 142. An electronically controllable switch 150 **partitions bus** 120 into **bus segment** 142 and a **bus segment** 152. Each of the switches 130, 140 and 150 includes a switch element for switching...transaction are closed, and the remaining switches, if any, are left open. As a result, **bus segments** required for performing the **bus** transaction are interconnected, and the remaining **bus segments** are electrically isolated from the interconnected **bus segments** . This reduces power consumption and capacitance. For example, assume that a bus transaction involves transfer...

...1 12. In this case, the switch elements of switch 130 are closed, thereby interconnecting **bus segments** 132 and 134, and the switch elements of switch 140 are opened, thereby electrically isolating **bus segments** 142 and 152 from **bus segments** 132 and 134. Accordingly, the **bus** transaction is performed on **bus segments** 132 and 134, and the capacitance of **bus segments** 142 and 152 is removed from the **bus** for this transaction, thereby reducing the power dissipation associated with the bus transaction. In this...

...transaction is performed between two blocks (blocks II 6 and II 8) on a single **bus segment** (**segment** 152), and the remainder of **bus** 120 is isolated from **bus segment** 152.

Some **bus** transactions may involve more than one destination. In that case, the appropriate switches are closed to interconnect the source functional block to the destination functional blocks, and any unneeded **bus segments** are isolated by opening the appropriate switches. For example, where functional block I 1 0...

...Source and destination information is present in the electronic circuit in order to control each **bus** transaction, even in the absence of **bus partitioning** . For example, control signals may identify the **bus** master (source), and address signals may identify the destination. From the source and destination information...

...destination pairs. The switch controller 160 controls the states of the switches 130, 140, 150 **dynamically** during the sequence of **bus** transactions to ensure proper interconnection of **bus segments** to enable the **bus** transactions to be performed. Thus, the switch controller 160 **dynamically** changes the states of the switches between **bus** transactions. Latch 164 ensures that the switch states are stabilized when each bus transaction is...

...within the scope of the invention. The switch configuration depends on the bus topology, the **bus** length and the added circuitry needed for **bus partitioning** . Examples of different switch configurations for an electronic apparatus having three functional blocks are shown...

...220. In the example of Fig. 3, switches 230, 232 and 234 are placed in **bus segments** connected to functional blocks 210, 212 and 214, respectively. In this configuration, at least two...

...perform a bus transaction. In Fig. 4, switches 240 and 244 are placed in the **bus segments** connected to ftinctional blocks 2 1 0 and 214, respectively. The switches 240 and 244 can be positioned in any two of the **bus segments** . In Fig. 5, a single switch 250 is placed in the **bus segment** connected to functional block 21 0. This configuration may be desirable, for example, where the **bus segment** connected to functional block 210 is longer than the other **bus segments** . The single switch 250 may be positioned in any of the **bus segments** as desired. Fig. 6 illustrates a switch configuration where a switch 260 is

placed. ...switch. Figs. 3-6 illustrate the fact that many switch configurations may be utilized for **partitioning** a **bus** , even where the **bus** is connected to only three functional blocks.

Bus partitioning depends on a number of factors, including **bus** length and **bus** geometry. The optimal **partitioning** from a speed standpoint is to **partition** the **bus** into equal **bus segments** , assuming that **wire** capacitance dominates gate capacitance. The RC delay of the **wire** increases quadratically with the **wire** length. As the **bus** is **partitioned** into more **segments** , the **wire** delay goes down,. while the driver delay goes up, because some delay is added by...

...the functional blocks that communicate most frequently with each other are located on the same **bus segment** . This permits a larger percentage of the capacitance to be isolated from the bus more frequently. A given **bus** may be **partitioned** into 2 to n **segments** , and the **segments** may be interconnected by I to m electronically controllable switches. A particular **bus** transaction may require I to n **segments** and may require that 0 to m switches be closed.

An example of an implementation of a **dynamic** electronically controllable switch for **bus partitioning** is shown in Fig. 7. Circuitry for a single **bus conductor** is shown in Fig. 7. It will be understood that the circuitry is repeated for each **bus conductor** . **Dynamic** logic is logic that has a precharge phase and an evaluation phase, both controlled by...

...phase.

Precharge circuits 310, 312 and 314 receive the system clock and are connected to **bus segment** I **conductor** 320, **bus segment** 2 **conductor** 322 and **bus segment** 3 **conductor** 324, respectively. A NOR gate 330 and an FET 332 are connected between **segment** 1 **conductor** 320 and **segment** 2 **conductor** 322. NOR gate 330 receives an enable signal En(O) from switch controller 160 (Fig. 2). A NOR gate 334 and an FET 336 are connected between **segment** I **conductor** 320 and **segment** 3 **conductor** 324. NOR gate 334 receives an enable signal En(1) from switch controller 160. A NOR gate 340 and an FET 342 are connected between **segment** 2 **conductor** 322 and **segment** 1 **conductor** 320. NOR gate 340 receives an enable signal En(2) from switch controller 160. A NOR gate 344 and an FET 346 are connected between **segment** 2 **conductor** 322 and **segment** 3 **conductor** 324. NOR gate 344 receives an enable signal En(3) from switch controller 160. A NOR gate 350 and an FET 352 are connected between **segment** 3 **conductor** 324 and **segment** I **conductor** 320. NOR gate 350 receives an enable signal En(4) from switch controller 160. A NOR gate 354 and an FET 356 are connected between **segment** 3 **conductor** 324 and **segment** 2. **conductor** 322. NOR gate 354 receives an enable signal En(5) from switch controller 160. Each...

...unidirectional switch element 362.

The switch embodiment of Fig. 7 provides a directional connection between **bus segment conductors** . Thus, for example, switch element 360 provides a connection in one direction from **segment** I **conductor** 320 to **segment** 2 **conductor** 322; and switch element 362 provides a connection in the reverse direction from **segment** 2 **conductor** 322 to **segment** I **conductor** 320. Thus, switch elements 360 and 362 constitute a bi-directional switch element of an electronically controllable switch for **partitioning bus segments** 1 and 2.

An example of an implementation of a static electronically controllable switch for **bus partitioning** is shown in Fig. 8. Like elements in Figs. 7 and 8 have the same...

...not dependent -loon a clock for proper evaluation. A unidirectional switch element 400, connected between **segment** I **conductor** 320 and **segment** 2 **conductor** 322, includes a NAND gate 4 1 0, a NOR gate 412 and FETs 414 and 416. NAND gate 41 0 receives the **bus** signal on **segment** I **conductor** 320 and an enable signal En(l). The output of NAND gate 410 is connected to the gate of FET 414. NOR gate 412 receives the **bus** signal on **segment** I **conductor** 320 and an inverted enable signal En-n(l). The output of NOR gate 412...

...the supply voltage and ground. The node connecting FETs 414 and 416 is connected to **segment** 2 **conductor** 322. A unidirectional switch element 418, including NAND gate 420, NOR gate 422 and FETs 424 and 426, is used for connecting **segment** 2 **conductor** 322 to **segment** I **con**ductor 320.

Unidirectional switch elements 400 and 418 constitute a bidirectional switch element of an electronically controllable switch for **partitioning bus segments** I and 2. This circuit is replicated for each switch element of each switch.

The...

...2, 114 and I 1 6, and switches 13 0, 140 and 15 0 for **partitioning bus** 120. A switch controller 5 1 0 includes a first source-destination decoder 520, a...

...requiring switch 150 to be closed. In this case, switch 140 is open. The first **bus** transaction is performed on a first **bus section** including **bus segments** 132 and 134, and the second **bus** transaction is performed on a second **bus section** including **bus segments** 142 and 152. Switch 140 electrically isolates the **bus sections** such that the first and second **bus** transactions can be performed simultaneously.

The ability to perform simultaneous bus transactions depends on the bus configuration and on the transactions to be performed. In many instances, two **bus** transactions cannot be performed simultaneously, because the same **bus segment** is required for both **bus** transactions. However, when **bus** transactions can be performed simultaneously, higher throughput is achieved, and the power dissipation associated with...
...that the present invention is not limited to two simultaneous bus transactions. Two or more **bus** transactions can be performed simultaneously, if permitted by the **bus partitioning** topology and the required transactions. This feature may be particularly advantageous in the case of...

Claim
1 Electronic apparatus comprising:
a plurality Of functional electronic blocks;
a **bus** interconnecting said functional blocks;
one or more electronically controllable switches **partitioning** said **bus** into **bus segments** , each of said electronically controllable switches having an on state wherein two of said **bus segments** are interconnected and an off state wherein said two **bus segments** are isolated; and
a switch controller for controlling the states of said electronically controllable switches...

...single substrate.

5 Electronic apparatus as defined in claim 1, wherein said bus comprises multiple **conductors** and wherein each of said electronically controllable switches **partitions** each **conductor** of said **bus** .

6 Electronic apparatus as defined in claim 1, wherein said switch controller comprises a source...

...by a bus, a method for communicating between the functional blocks, comprising the steps of
partitioning the **bus** into **bus segments** ;
enabling a **bus** transaction between a source functional block and a destination ftinctional block by interconnecting **bus segments** to complete a connection between the
source and destination ftinctional blocks; and
performing the **bus** transaction on the interconnected **bus segments** .

12 A method as defined in claim I 1, wherein the step of **partitioning** the **bus** comprises **partitioning** a **bus** fabricated on a single substrate.

13 A method as defined in claim I 1, wherein the step of **partitioning** the **bus** comprises **partitioning** the **bus** with one or more electronically controllable switches and wherein the step of enabling a bus...

...bus transaction.

14 A method as defined in claim I 1, wherein the step of **partitioning** the **bus** comprises **partitioning** the **bus** with one or more electronically controllable switches and wherein the step of enabling a **bus** transaction comprises **dynamically** changing the states of said electronically controllable switches between transactions of a sequence of bus...

...bus transactions.

15 A method as defined in claim I 1, wherein the step of **partitioning** the **bus** comprises **partitioning** the **bus** with one or more electronically controllable switches and wherein the step of enabling a bus...

...16 A method as defined in claim 1 1, wherein the step of enabling a **bus** transaction further comprises the step of electrically isolating **bus segments** that are not needed for the **bus** transaction.

17 Electronic apparatus comprising:
a plurality of functional **electronic** blocks **interconnected** by a **bus** ;
means for **partitioning** the **bus** into **bus segments** ; and
means for enabling a **bus** transaction on the **bus** by interconnecting **bus segments** in response to control information representative of the source and the destination of the bus...

...a single substrate.

21 Electronic apparatus as defined in claim 17, wherein said means for **partitioning** the **bus** comprises one or more electronically controllable switches, each having an on state and an off...